



1999

Java and Microsoft: How Does the Antitrust Story Unfold

Daniel J. Gifford

Follow this and additional works at: <https://digitalcommons.law.villanova.edu/vlr>



Part of the [Antitrust and Trade Regulation Commons](#)

Recommended Citation

Daniel J. Gifford, *Java and Microsoft: How Does the Antitrust Story Unfold*, 44 Vill. L. Rev. 67 (1999).
Available at: <https://digitalcommons.law.villanova.edu/vlr/vol44/iss1/2>

This Article is brought to you for free and open access by Villanova University Charles Widger School of Law Digital Repository. It has been accepted for inclusion in Villanova Law Review by an authorized editor of Villanova University Charles Widger School of Law Digital Repository.

1999]

JAVA AND MICROSOFT: HOW DOES THE ANTITRUST STORY UNFOLD?

DANIEL J. GIFFORD*

TABLE OF CONTENTS

I. INTRODUCTION	67
II. THE VISION OF THE JAVA PLATFORM	72
A. <i>The Government and Microsoft</i>	72
B. <i>The Java Platform Paradigm: A Vision of the Future?</i>	74
C. <i>The Technology</i>	81
1. <i>Operating Systems</i>	81
2. <i>The Internet and the World Wide Web</i>	84
3. <i>Browsers</i>	85
4. <i>Java</i>	87
5. <i>Java Rivals</i>	89
III. PATH DEPENDENCY, NETWORK EFFECTS AND THE DEVELOPMENT OF COMPUTER SOFTWARE	90
A. <i>The Economics</i>	90
B. <i>Historical Events Contributing to Microsoft's Dominance Over Operating Systems</i>	93
IV. THE FUTURE OF THE SOFTWARE INDUSTRY: THE GOVERNMENT CASE AND THE JAVA PLATFORM PARADIGM	96
A. <i>Tying the Browser to the Operating System</i>	97
B. <i>Innovation and Incompatible Systems</i>	110
V. HOW INNOVATION AND INCOMPATIBILITY RELATE TO THE JAVA PLATFORM PARADIGM	111
A. <i>Two Kinds of Incompatibility</i>	112
1. <i>Innovation and the Process of Standardization in a Context of No-Holds Barred Competition</i>	112
2. <i>Innovation Implicating the Java Platform Paradigm</i>	116
B. <i>Antitrust, Innovation and the Java Platform Paradigm</i>	122
VI. CONCLUSION	123

I. INTRODUCTION

RAPID developments in the software industry underlie both the Government's antitrust proceedings against the Microsoft Corporation¹ and the lawsuit brought by Sun Microsystems, Inc. ("Sun") against

* Robins, Kaplan, Miller & Ciresi Professor of Law, University of Minnesota. The author acknowledges helpful comments on an earlier draft of this article from Jim Chen, Paul Edelman, Daniel A. Farber, Peter Klausler, David McGowan, John Risken, James J. Risser, Thomas Smith and E. Thomas Sullivan.

1. See generally Complaint, United States v. Microsoft Corp., 1998-2 Trade Cas. (CCH) ¶ 72,261 (D.D.C. Sept. 14, 1998) (filed May 18, 1998) (No. 98-1232) [here-

Microsoft in which Sun is asserting claims grounded in breach of contract, trademark infringement and unfair competition.² In October 1997, the Government challenged the Microsoft Corporation's right to bundle its internet browser (the Microsoft Internet Explorer 4.0) with its Windows 95 operating system.³ That proceeding was brought as a request for an order to show cause why Microsoft should not be held in contempt for violating a consent decree ending a suit brought by the Government in 1995, the primary focus of which was on Microsoft's licensing practices. In May 1998, the Government instituted a new action against Microsoft. This new action again challenged Microsoft's bundling of its browser with its operating system, including its new Windows 98 operating system.⁴ In the new action, the Government also challenged the tie as monopolization and attempted monopolization.⁵ In addition, the Government added other claims, including an alleged attempt to divide the browser market with Netscape⁶ and several agreements with internet services providers and internet content providers,⁷ matters which are not discussed in this Article. Although Microsoft sees the browser as "integrated" into its operating system so as to form one product, the Government sees the browser and the operating system as two products "tied" together. Because the computer codes underlying the browser and operating system overlap even more in the Windows 98 context than in Windows 95,⁸ the Government's burden in the new antitrust case on that issue will probably be heavier than it bore in the contempt proceeding.

The Government wants Microsoft to disentangle the browser from its Windows 98 and Windows 95 operating systems. Indeed, not far removed from both the contempt proceeding and the new antitrust suit are issues connected with Microsoft's modifications of Java, Sun Microsystems's recently developed programming language and virtual machine architec-

inafter Complaint]; Contempt Petition, *United States v. Microsoft Corp.*, 1997 WL 656528 (D.D.C. Oct. 20, 1997) (No. 94-1564) [hereinafter Contempt Petition].

2. See John Markoff, *Sun Sues Microsoft on Use of Java System*, N.Y. TIMES, Oct. 8, 1997, at D4 [hereinafter Markoff, *Sun Sues Microsoft*] (describing lawsuit between Sun Microsystems and Microsoft over Java programming language); see also Amended Complaint at 17-20, *Sun Microsystems, Inc. v. Microsoft Corp.*, 999 F. Supp. 1301 (N.D. Cal. 1998) (No. 97-20884) [hereinafter Amended Complaint], available in <<http://java.sun.com/lawsuit/complaint.html>> (alleging breach of contract, trademark infringement and unfair competition).

3. See generally Contempt Petition, *supra* note 1 (alleging that Microsoft had violated terms of consent decree by bundling its browser with its operating system). The consent decree had been entered in August 1995. See *United States v. Microsoft Corp.*, 1995-2 Trade Cas. (CCH) ¶ 71,096 (D.D.C. 1995).

4. See Complaint, *supra* note 1, ¶¶ 103-23, 134-37 (alleging tying of Microsoft's browser software to its operating system).

5. See *id.* ¶¶ 138-41.

6. See *id.* ¶¶ 70-71.

7. See *id.* ¶¶ 75-92.

8. In Windows 98, the browser is fully integrated into the operating system. See, e.g., John Montgomery, *The Next Windows*, BYTE, Jan. 1998, at 56.

ture.⁹ Sun is litigating Microsoft's right under its license from Sun to introduce its own modifications of Java programming. Underlying both the Sun/Microsoft lawsuit and the Government's continuing antitrust challenges are competing visions of the future of personal computing and the internet. In hearings before the Senate Judiciary Committee, the Chief Operating Officers of Sun and Netscape complained that Microsoft was abusing its dominance in the personal computer operating systems market.¹⁰ Sun and Netscape have a different vision of the future than does Microsoft.

These disputes involve, in one way or another, a vision of the world of business and personal computing that is significantly different from the present. This vision is that of the Java platform paradigm—one in which browsers employing Java programming and connected via the internet to servers manipulate software applications programs residing on servers rather than on the hard disk of the computer user. This vision is a powerful one. Its realization would carry enormous economic consequences, including, inter alia, the undermining of Microsoft's dominance over personal computer operating systems. It is a vision that the Government has invoked directly in its new antitrust lawsuit.¹¹

This Article explores the connections between this vision and the issues of antitrust tying and monopolization surrounding Microsoft's relations to Java and the browser wars. It describes the relevant technology and its consequences—how the software industry has largely settled around a dominant operating system, the Windows platform,¹² and how

9. See David Bank & John R. Wilkie, *Microsoft Probe by U.S. Is Expanded to Cover Issues Related to Sun's Java*, WALL ST. J., Mar. 17, 1998, at B6.

10. See generally *Market Power and Structural Change in the Software Industry: Hearing Before the Senate Comm. on the Judiciary*, 105th Cong., 2d Sess. (1998), available in 1998 WL 8992527 (testimony of Scott McNealy, President and CEO, Sun Microsystems, Inc., ("Sun") and testimony of Jim Barksdale, President and CEO, Netscape Communications, Inc. ("Netscape")) [hereinafter *Senate Hearing*]. Netscape had previously complained to the Justice Department of Microsoft bundling and other behavior allegedly violative of the antitrust laws. See *Netscape Letter to Justice Department Accuses Microsoft of Antitrust Violations*, Antitrust & Trade Reg. Daily (BNA) (Aug. 23, 1996).

11. See Complaint, *supra* note 1, ¶¶ 8, 68, 73, 107, 122 (containing allegations based on Java platform paradigm).

12. See Memorandum of the United States in Support of Petition for an Order to Show Cause Why Respondent Microsoft Corporation Should Not Be Found in Civil Contempt at 5 n.1, *United States v. Microsoft Corp.*, 1997-2 Trade Cas. (CCH) ¶ 71,990 (D.D.C. 1997) (No. 94-1564) [hereinafter *Memorandum*] (explaining that in computer industry jargon, operating systems, in combination with underlying hardware, have come to be referred to as "platforms"). Operating systems serve as a "platform" (containing the applications programming interfaces or "APIs") on top of which the applications run and for which they are written. See *id.* at 2; see also WILLIAM S. DAVIS, *OPERATING SYSTEMS: A SYSTEMIC VIEW* 3 (4th ed. 1992). Application software (such as word-processing programs, spreadsheet programs, data management programs, etc.) thus is written for a particular platform—a Windows platform, a Macintosh platform, a UNIX platform, etc. Although Java is not an operating system, it nonetheless controls how a developer

that dominance may be threatened by the advent of Java programming and the Java platform. This Article also discusses some of the problematic aspects of the vision. It also addresses the relation of the concepts of network externalities and path dependency to the emergence of the Java-based challenge to the Windows platform. This discussion identifies factors that promise to advance or retard the potential replacement of the Windows platform by a Java substitute and factors that may only appear to do so.

Although the Java platform paradigm plays a role in Sun's lawsuit, the Government's proceedings squarely bring the Java paradigm into the anti-trust context.¹³ Both of the Government's antitrust proceedings raise issues that are important and not well understood by many antitrust practitioners. Although the contempt proceeding has been fought largely under the rubric of antitrust "tying" doctrine¹⁴ and the new antitrust case asserts claims under monopolization, attempted monopolization and tying, the issues of concern have to do with the way antitrust law interacts with a highly dynamic modern industry in the throes of change. In the past, a number of commentators have suggested that where network externalities and path dependency analysis become relevant, an aggressive application of antitrust becomes warranted.¹⁵ Although these theories explain much of Microsoft's dominance in operating systems, their application to the browser wars needs more analysis than has appeared in the literature to this point. The Government may view its lawsuits as preemptive strikes designed to prevent or to impede Microsoft from using network effects to advance down a path towards browser dominance.¹⁶ The vision of the Java platform paradigm plays a supporting role in the Government's cases. Sun's lawsuit, while raising legally mundane issues of con-

writes applications software analogous to the way an operating system controls how that software is written. See generally Mark A. Lemley & David McGowan, *Could Java Change Everything? The Competitive Propriety of a Proprietary Standard*, 43 ANTITRUST BULL. 715, 749 (1998) [hereinafter Lemley & McGowan, *Competitive Propriety*] (explaining how Java might transform browser into platform-independent meta-operating system). Some observers refer to a Java platform as analogous to the Windows platform. See *id.* This Article adopts that usage.

13. For a discussion of the Java platform paradigm in the context of the Government's new antitrust lawsuit, see *supra* note 11 and accompanying text.

14. See generally *United States v. Microsoft Corp.*, 147 F.3d 935 (D.C. Cir. 1998) (holding browser integrated into operating system).

15. See, e.g., Kenneth C. Baseman et al., *Microsoft Plays Hardball: The Use of Exclusionary Pricing and Technical Incompatibility to Maintain Monopoly Power in Markets for Operating System Software*, 40 ANTITRUST BULL. 265, 298-315 (1995) (arguing that presence of network externalities and path dependency requires more stringent antitrust enforcement).

16. See Lemley & McGowan, *Competitive Propriety*, *supra* note 12, at 718-21, 726-27, 734-35 (relating network effect analysis to Government's current intervention). Microsoft's tie of its browser to its operating system produces anticompetitive consequences over time through network effects.

tract interpretation, trademark law and unfair competition, also invokes the Java platform paradigm for support.¹⁷

In attempting to untangle the various disputes and the ways that they relate to the Java platform paradigm, this Article deals with a number of related issues. Part II describes the vision of the Java platform paradigm as a potential successor to the Windows platform paradigm.¹⁸ Part II also briefly describes some of the technology forming the background to the scenario in which Java replaces Windows as the dominant platform.¹⁹ Part III reviews the concepts of network externalities and path dependency.²⁰ Part III also considers how these concepts relate to Microsoft's present dominance in personal computer operating systems.²¹ Part IV deals directly with the Government's assertion that Microsoft is unlawfully tying its browser to its operating system and thereby violating section one or two of the Sherman Act.²² Also in Part IV, this Article concludes that while the Government's contention is certainly doctrinally plausible, it is nonetheless a highly technical one with a number of problematic aspects.²³ The Government's real rationale in pursuing its tying and related monopolization claims lies in its apparent belief that this tie is aiding Microsoft to protect its Windows platform monopoly against technological changes that threaten to undermine it.²⁴ Part V addresses the way innovation has been affected by, or is related to, the browser wars, the Sun/Microsoft dispute and the Government's antitrust actions.²⁵ Part V also explores some of the industry developments most relevant to the Government's the-

17. See Plaintiffs' Second Amended and Supplemental Complaint ¶¶ 48-54, *Sun Microsystems, Inc. v. Microsoft Corp.*, 21 F. Supp. 2d 1109 (N.D. Cal. 1998) (filed May 12, 1998) (No. 97-20884) [hereinafter Second Amended Complaint] (alleging that Java platform paradigm threatens Microsoft's dominance in operating systems).

18. For a discussion of the vision of the Java platform paradigm as a successor to the Windows platform paradigm, see *infra* notes 43-83 and accompanying text.

19. For a discussion of the technology forming the background to the scenario in which Java replaces Windows as the dominant platform, see *infra* notes 84-150 and accompanying text.

20. For a discussion of the concepts of network externalities and path dependency, see *infra* notes 151-65 and accompanying text.

21. For a discussion of the concepts of network externalities and path dependency and their relation to Microsoft's present dominance in the operating system market, see *infra* notes 166-85 and accompanying text.

22. 15 U.S.C. §§ 1-7 (1994). For a discussion of the Government's assertion that Microsoft is unlawfully tying its browser to its operating system, see *infra* notes 189-265 and accompanying text.

23. For a discussion of conclusions regarding the Government's contention regarding Microsoft, see *infra* notes 266-69 and accompanying text.

24. For a discussion of technological innovations and incompatible systems, see *infra* notes 266-69 and accompanying text.

25. For a discussion of the effects that the browser wars, the Sun/Microsoft dispute and the Government's antitrust action have had upon innovation in the software industry, see *infra* notes 270-323 and accompanying text.

ory.²⁶ Finally, this Article concludes that the potential for the erosion of the Microsoft monopoly is dependent upon technological developments; that the belief by the Government, Sun and some commentators that the advent of the Java programming language is likely to accomplish that result is problematic; and that the future of the Windows platform is likely to be determined by the market, regardless of (or in spite of) government antitrust initiatives.²⁷ The concept of network externalities, far from working in favor of Microsoft technology as it did with operating systems, is working against Microsoft Windows-only internet technologies. This Article argues that if Microsoft technologies (such as J/Direct and ActiveX) ultimately win in the market, they will prevail not because of network effects, but in spite of them. For these technologies to prevail, they must be sufficiently superior to overcome the network value represented by Java.²⁸

II. THE VISION OF THE JAVA PLATFORM

A. *The Government and Microsoft*

The contempt proceeding grew out of Microsoft's efforts to bundle its browser with its operating system. The later antitrust suit also challenges this bundling.²⁹ Microsoft has been insisting that computer manufacturers who purchase licenses to equip their products with the Windows 95 operating system also include its browser, the Internet Explorer, on those machines. Microsoft's Windows 98 operating system increases the integration of the browser into the operating system.³⁰

The current challenges are the latest in a series that the Government has made—or has considered making—to Microsoft.³¹ As explained later, the Government, in most or all of these actions, has been heavily influenced by the theory of network externalities.³² In 1995, the Government thwarted Microsoft's attempt to acquire the Intuit Corporation and its financial planning software, Quicken. As a result of the Government's challenge, the parties decided to call off their merger.³³ Later that year, the

26. For a discussion of some of the industry developments most relevant to the Government's theory, see *infra* notes 290-321 and accompanying text.

27. For a discussion of this Article's conclusions, see *infra* notes 270-323 and accompanying text.

28. For a discussion of the relation of the concept of network externalities to the Java platform paradigm, see *infra* notes 151-85, 322-23 and accompanying text.

29. See Complaint, *supra* note 1, ¶¶ 103-23, 134-37 (alleging tying of Microsoft's software to its operating system).

30. See Montgomery, *supra* note 8, at 57.

31. See Complaint, *supra* note 1, ¶¶ 42-52 (listing chronology of United States's antitrust challenges related to current lawsuit); see also *A History of the Microsoft Case*, AUSTIN AMERICAN-STATESMAN, May 19, 1998, at D8, available in 1998 WL 3611045 (same).

32. For a discussion of the influence of the theory of network externalities on the Government's actions, see *infra* notes 220-24 and accompanying text.

33. See Daniel J. Gifford, *Microsoft Corporation, the Justice Department, and Anti-trust Theory*, 25 Sw. U. L. REV. 621, 657-65 (1996) [hereinafter Gifford, *Microsoft*].

Government considered challenging Microsoft's inclusion, on its Windows 95 operating system, of an icon enabling computer buyers to subscribe to the Microsoft Exchange, Microsoft's then new proprietary network with internet access.³⁴ In 1994, the Government challenged Microsoft's licensing practices, practices that were equivalent to exclusive supply contracts.³⁵ In their context, the lawfulness of those licensing practices was unclear.³⁶ That action ended in a consent decree.³⁷ That consent decree was the basis for the Government's first challenge to the browser/operating system bundle. The bundle has since been further challenged in an entirely new antitrust suit.

The new antitrust suit alleges that Microsoft is violating the antitrust laws by unlawfully tying its browser to the two operating systems (Windows 95 and Windows 98).³⁸ It does so both in the language of tying doctrine and in the language of monopolization and attempted monopolization.³⁹ In its earlier challenge under the consent decree, the Government contended that Microsoft was violating the consent decree by tying its browser to its operating system.⁴⁰ In the consent proceeding, the district court ruled that the consent decree was sufficiently ambiguous as to preclude its punishing Microsoft for its past behavior, but temporarily enjoined Microsoft from requiring manufacturing licensees to include the browser on new computers.⁴¹ That determination, however, was reversed on appeal when the appellate court took the view that on the record before the

Corporation] (detailing Microsoft's planned acquisition of Intuit, Inc.); *Microsoft and Intuit Abandon Merger Challenged by Division*, 68 ANTITRUST & TRADE REG. REP. 672, 672 (1995) (reporting on Microsoft's called-off \$2 billion acquisition of Intuit, Inc.).

34. See Gifford, *Microsoft Corporation*, *supra* note 33, at 665-69 (discussing antitrust issues related to location of Microsoft Exchange icon on Windows 95).

35. See *id.* (describing 1994 antitrust action and subsequent consent decree); see also John E. Lopatka & William H. Page, *Microsoft, Monopolization, and Network Externalities: Some Uses and Abuses of Economic Theory in Antitrust Decision Making*, 40 ANTITRUST BULL. 317, 326-28 (1995) [hereinafter Lopatka & Page, *Antitrust Decision Making*] (noting Justice Department's characterization of licensing practices as "requirements contracts").

36. See Gifford, *Microsoft Corporation*, *supra* note 33, at 631-44 (discussing lawfulness of subject licensing practices).

37. See *United States v. Microsoft Corp.*, 1995-2 Trade Cas. (CCH) ¶ 71,096 (D.D.C. 1995) (issuing consent decree).

38. See Complaint, *supra* note 1, ¶¶ 103-23, 134-37 (alleging that Microsoft unlawfully tied its internet browser to its Windows 95 and Windows 98 operating systems).

39. See *id.* The tie of the browser to the operating system is challenged as a violation of section one of the Sherman Act. See *id.* ¶¶ 134-37. It is also challenged as monopolization and attempted monopolization. See *id.* ¶¶ 138-41.

40. See Memorandum, *supra* note 12, at 15 (charging that Microsoft violated consent decree by tying internet browser to its operating systems).

41. See *United States v. Microsoft Corp.*, 980 F. Supp. 537, 541 (D.D.C. 1997) (concluding, *inter alia*, that Microsoft demonstrated sufficiently ambiguity of term "integrated product" under consent decree, but construing consent decree and enjoining tie), *rev'd*, 147 F.3d 935 (D.C. Cir. 1998).

court there was no tie because the browser and the operating system were one product.⁴²

B. *The Java Platform Paradigm: A Vision of the Future?*

The theory underlying the Government's current initiative is a complex one. The Government believes that the Java programming language is threatening to undermine Microsoft's present dominance over operating systems and that Microsoft is trying to forestall that threat by attempting to control internet technology.⁴³ The Government believes that a major step in Microsoft's attempt to control internet technology is to seek dominance in browsers. The Government further believes that Microsoft is using its dominance of personal computer operating systems as a lever to achieve browser dominance.⁴⁴

Microsoft now holds a practical monopoly over personal computer operating systems. Over eighty percent of personal computers run on some version of Microsoft's Windows.⁴⁵ Older personal computers generally run on Microsoft's MS-DOS operating system with a Windows 3.1 overlay.⁴⁶ Almost all new machines equipped with an Intel microprocessor (or a simulation) and designed for the consumer market are equipped at the factory with the Windows 98 operating system.⁴⁷ Many business machines run on Windows for Workgroups and Windows NT.⁴⁸ Because Microsoft's Windows systems dominate the market, application software developers possess strong economic incentives to write primarily for Windows systems.⁴⁹ As a result, more applications programs exist for Windows platforms than for alternative platforms like Macintosh, OS/2, Linux, UNIX

42. See *Microsoft Corp.*, 147 F.3d at 948-52 (concluding that Windows 95 and Microsoft's internet browser do not exist separately and are functionally single product).

43. See Complaint, *supra* note 1, ¶¶ 8, 37, 68, 73, 107, 122 (alleging that Microsoft is threatened by Java and that Microsoft is attempting to control internet technology).

44. See *id.* ¶ 13 ("Microsoft's conduct with respect to browsers is a prominent and immediate example of the pattern of anticompetitive practices undertaken by Microsoft with the purpose and effect of maintaining its [personal computer (PC)] operating system monopoly and extending that monopoly to other related markets.").

45. See *id.* ¶¶ 2, 57 (alleging that Windows operating systems are used on over 80% of Intel-based PCs).

46. See Gifford, *Microsoft Corporation*, *supra* note 33, at 625 (noting that original Windows was overlay on underlying MS-DOS operating system).

47. See Complaint, *supra* note 1, ¶ 2 ("More than 90% of new Intel-based PCs are shipped with a version of Windows pre-installed. PC manufacturers . . . have no commercially reasonable alternative to Microsoft operating systems for the PCs that they distribute.").

48. See Elizabeth Corcoran, *Microsoft Says New Product to Be Major; Windows NT 5.0 Will Be Core of Its Systems*, WASH. POST, July 24, 1998, at F3 (noting that Windows NT has many business users).

49. See Tom R. Halfhill, *TODAY the WEB, TOMORROW the WORLD*, BYTE, Jan. 1997, at 68, 70 [hereinafter Halfhill, *TODAY the WEB*] (stating that many commer-

and others.⁵⁰ This surfeit of applications software for Windows helps to reinforce Microsoft's dominance in operating systems.

The development by Sun Microsystems of the Java programming language threatens Microsoft's present hegemony. Java possesses a number of remarkable capabilities, one of which is interoperability.⁵¹ A program written in Java is capable of running on any platform possessing a Java interpreter or a Java virtual machine (*i.e.*, software that converts the program from Java code into the machine code specific to the user's platform). This capability allows a program written in Java to run on numerous platforms.⁵² The same software thus can be marketed to the Windows, Macintosh, OS/2, Linux and UNIX segments of the personal computer market.⁵³

At the present time, the focus of Java's ramifications is with the internet. Programs written in Java and placed on server websites can be received by users regardless of the operating system running the user's machine.⁵⁴ Currently, many websites contain small applications programs written in Java (Java applets) that can interact with a viewer or otherwise provide the viewer with screen-based action, such as animations, or minor services, such as the computation of mortgage payments or the analysis of other financial data based upon the user's input.⁵⁵ Java carries the promise, however, of accomplishing vastly more than visual or other incidental enhancement of websites because, as a programming language, Java can provide the code for sophisticated applications programs.⁵⁶ Java's inter-

cial developers write their software for Windows first because Windows runs on about 90% of world's PCs).

50. *See id.* (noting that Macintosh, OS/2 and UNIX are minority platforms, while Microsoft continues to dominate).

51. *See* Richard V. Dragan & Larry Seltzer, *Java: A Field Guide for Users*, PC MAG., May 27, 1997, at 100, 100. New operation system developers face a significant obstacle in trying to break into the market because most software developers write first for Microsoft's Windows. *See* Halfhill, *TODAY the WEB*, *supra* note 49, at 71 (noting that commercial developers write software for Windows first).

52. *See* Halfhill, *TODAY the WEB*, *supra* note 49, at 72 (noting that "Cross-platform" compatibility is large factor in Java's early success). Some software companies have already begun to write Java development tools so that their programs will run on any operating machine. *See id.*

53. *See id.* (discussing "write-once, run anywhere" character of programs written in Java).

54. *See id.* (noting that Java programs can run independent of operating system being used). One computer software developer has stated that Java represents the first language in which "developers can write applications using their Windows people, their UNIX people, and their Mac[intosh] people." *Id.*

55. *See* Dragan & Seltzer, *supra* note 51, at 101 (noting that most common use of Java is creation of interactive web page animated images).

56. *See* Halfhill, *TODAY the WEB*, *supra* note 49, at 70 (discussing new characteristics of Java that "help overcome [its] reputation as a lightweight language"). A forerunner of a new breed of sophisticated applications programs is Lotus JavaBean-based eSuite. *See* Steve Gillmor, *A Different Desktop for the Workplace*, BYTE, Feb. 1998, at 41, 41; David Strom, *Java-based App Shows Promise But Needs Work*, COMPUTERWORLD, Feb. 9, 1998, at 53, 53.

operability feature allows programs to be available to users from internet servers, regardless of the differing platforms employed by those users.⁵⁷ Users will access the software by tying into the server. Rather than being limited to the software installed on their hard disks, users will have access to an infinite variety of applications programs located on servers.⁵⁸

Currently, most marketed versions of internet browsers are able to handle Java programs.⁵⁹ These browsers incorporate Java virtual machines that translate programs in Java code into the machine code appropriate for the particular platform being employed by the browser user.⁶⁰ As the installed base of Java virtual machines grows, the incentives for internet servers to employ Java programming will increase.

At the moment, Java provides internet servers the ability to enhance their websites with visually dynamic decorative effects and the capability to interact with users at the level of simple computations. In the next stage of internet development, servers will carry sophisticated applications programs available for users through their browsers. Internet servers will replace users' individual hard disks as the source of applications programs.⁶¹ There is a substantial likelihood that these server-based application programs will be written in Java because Java's interoperability feature makes the development of Java programming more cost-effective than the alternatives.⁶² Programming can be written once in Java and it will be available for users on all platforms.

These developments will provide the context for the elimination of Microsoft's dominance over personal computer operating systems.⁶³

57. See *id.* (discussing Java's cross-platform capabilities).

58. See Edward Mendelson, *The Java Office*, PC MAG., May 27, 1997, at 103 (noting that Java programs can be downloaded from servers whenever they are needed).

59. See Dragan & Seltzer, *supra* note 51, at 101 (stating that HotJava, Microsoft Internet Explorer, Netscape Communicator and Netscape Navigator have ability to handle Java programs).

60. See *id.* (stating that "Java byte codes can be executed on any system that has a Java virtual machine"). There appears to be grounds for believing that the Microsoft Explorer handles Java programs faster and more reliably than the rival Netscape Communicator 4.01. See Tom R. Halfhill & Al Gallant, *How to Soup Up Java*, BYTE, May 1998, at 60, 70 (comparing Microsoft Explorer's and Netscape Communicator 4.01's handling of Java programs); see also David Seltzer, *Editor's Choice: Microsoft JVM for Windows; Microsoft JVM for Windows is Named the Top Java Environment*, PC MAG., Apr. 7, 1998, at 79, 79 (commenting on speed of Microsoft JVM).

61. See Mendelson, *supra* note 58, at 103 (stating that Java programs will be downloaded from internet or server).

62. The cost of writing several versions of a program is saved when a program can be written once in Java and is usable by all platforms. Having to write different programs for different operating systems is "extremely time-consuming and expensive." Amended Complaint, *supra* note 2, ¶ 12.

63. See Halfhill, *TODAY the WEB*, *supra* note 49, at 70 (noting that Windows may become minority platform). The Government's memorandum to the district court in the consent proceeding embraced this vision:

When internet servers become the primary source for applications programs, the browser (rather than the Windows) interface will become the primary reference point for both software developers and computer users.⁶⁴ The ubiquitous Windows platform will be replaced by a Java-capable browser platform.⁶⁵ At that point, operating systems—while still needed—would become mere tools of the browser.⁶⁶ They would become interchangeable and Microsoft's power that it derives from Windows would wither away.⁶⁷ This vision is shared by the Government, Sun Microsystems, Netscape and others, and has received wide attention in business publications.⁶⁸ The attractiveness of this vision has also been recognized in the antitrust literature.⁶⁹ Netscape's new browser software,

The threat that competing browsers present to Microsoft's monopoly has two primary aspects: first, the degree to which browsers may become accepted as an *interface*; and second, the extent to which they can serve as a *platform* to which applications can be written that are independent of the underlying operating system.

Memorandum, *supra* note 12, at 34. The Government's reference to browsers serving as platforms for interoperable applications describes the vision articulated in the text. In its lawsuit against Microsoft, Sun includes references to this vision. See Amended Complaint, *supra* note 2, ¶¶ 51 & 52 (contending that Java's cross-platform capability poses serious threat to Microsoft's monopoly in operating systems market).

64. See Memorandum, *supra* note 12, at 32 (noting that as computer users increase their use of internet browsers as interfaces, they become less dependent on operating systems).

65. See Halfhill, *TODAY the WEB*, *supra* note 49, at 70 (stating that Java could "trigger the biggest platform switch since Windows surpassed DOS"). Apple, Microsoft, IBM and others recognize that Java's potential abilities will provide a "competitive advantage." See *id.* at 71 (discussing realization by several corporations that Java's potential and "superior run-time environment" will give them competitive advantages). According to the Government, Java "threatens to reduce or eliminate one of the key barriers to entry protecting Microsoft's operating system monopoly." Complaint, *supra* note 1, ¶ 7.

66. See Halfhill, *TODAY the WEB*, *supra* note 49, at 74 (noting that Java will only need to use operating systems as "hosts" to run its programs). Soon, Java virtual machines (VMs) will be available for most operating systems. See *id.* (predicting that there will be Java VMs for all major operating systems by end of 1997). One commentator noted that:

IBM is even porting Java to the AS/400 and MVS, which manage an estimated 70 percent of the world's corporate data. This is why it's not crazy to predict that by the turn of the century, there will be more copies of the Java VM in the world than any of the [operating systems] that host it.

Id.

67. See Complaint, *supra* note 1, ¶ 7 (contending that browser technology and Java will weaken Microsoft's operating system monopoly).

68. See, e.g., Amy Cortese et al., *The Software Revolution*, BUS. WK., Dec. 4, 1995, at 78, 78.

69. See, e.g., Lemley & McGowan, *Competitive Propriety*, *supra* note 12, at 734 (noting that Java's platform-independence may weaken "network benefits" Microsoft had enjoyed as result of operating system dominance); see also Mark A. Lemley, *Antitrust and the Internet Standardization Problem*, 28 CONN. L. REV. 1041, 1077 (1996) (discussing theory that Microsoft is trying to protect its operating sys-

which casts a shell over the Windows interface, is a step towards the implementation of this vision.⁷⁰

In the contempt proceeding, the Government argued that Microsoft was attempting to forestall this threat to its operating system monopoly by seeking to expand its share of the browser market and that part of that effort involved bundling its browser with its operating system.⁷¹ Although the Government ignored the several ways in which this tie differed from classic tying arrangements, the significance of these differences was overshadowed by Microsoft's alleged use of the tie as a means of strengthening monopoly power. The tie under challenge, therefore, was not a routine tying arrangement raising some doctrinally troublesome issues, but a critical component of Microsoft's efforts to maintain its Windows monopoly. In the new antitrust case, the Government has reasserted these tying claims, casting them in the language not only of section one, but also in the language of monopolization and attempted monopolization.⁷²

The Government, however, has not spelled out how Microsoft might maintain its Windows monopoly by expanding its share of the browser market. Microsoft's current browser, the Internet Explorer 4.0, carries Java capability.⁷³ Under the scenario described above, Java programming threatens Microsoft's Windows monopoly. As long as all of the major browsers carry Java virtual machines, however, shifts in the shares of the browser market among Netscape, Microsoft, Sun and others would leave Java's threat to Microsoft's Windows monopoly unaffected. Indeed, the Government's description of the Java-based scenario, under which the dominance of the Windows platform is eroded away, loses much of its explanatory power if the tie neither furthers nor impedes the progress of

tem monopoly from "new generation of computer operating systems integrated into Web browsers").

70. See Tom R. Halfhill, *Good-Bye, GUI Hello, NUI*, BYTE, July 1997, at 60, 66 [hereinafter Halfhill, *Good-Bye*] (claiming that Netscape's creation of full-screen desktop browser may make windows "irrelevant").

71. See *United States v. Microsoft Corp.*, 980 F. Supp. 537, 539 (D.D.C. 1997) (describing tying arrangement by which Microsoft exploits its operation systems monopoly), *rev'd*, 147 F.3d 935 (D.C. Cir. 1998). The district court observed that:

[T]he government charges that Microsoft coerces OEMs to license and distribute the Internet Explorer whether they want to or not, even though, the government asserts (and Microsoft vehemently denies), refusing to install Microsoft's browser will not affect the functioning of the underlying Windows 95 operating system in any other significant respect. The effect, says the government, is a classic "tying" arrangement by which Microsoft is exploiting its operation systems monopoly in violation of an express term of the Final Judgment.

Id.

72. See Complaint, *supra* note 1, ¶¶ 138-41.

73. See PAUL MCFEDRIES, *WINDOWS 95: UNLEASHED* 1242 (1998) [hereinafter MCFEDRIES, *WINDOWS 95*] (noting that Microsoft's current browser supports Java technology).

that scenario.⁷⁴ Perhaps in response to this potential missing link the Government has recently adverted to Microsoft's J/Direct technology, but the Government does not adequately develop the theory under which J/Direct either interferes with the emergence of the Java paradigm or is itself an anticompetitive innovation.⁷⁵

By contrast, Sun explicitly describes the how and why of Microsoft's response to the Java-based scenario.⁷⁶ After explaining why Java threatens Microsoft's dominance, Sun alleges that Microsoft has set out to undermine the promise of Java-based interoperability.⁷⁷ Under Sun's theory, Microsoft is introducing modifications to Java that interfere with Java's

74. See generally Memorandum, *supra* note 12, at 39 (acknowledging that United States cannot predict whether browsers will succeed in "commoditizing" operating systems and thereby diminish Microsoft's market power). Not only does the Government fail to connect the tie with the maintenance of the Microsoft monopoly, but the Government concedes that it has no objection to Microsoft ultimately dominating the browser market, so long as a browser/operating-system tie plays no role in its achieving that dominance. See *id.* (stating that its action is "intended only to ensure that the market remains able to choose between the competing technologies . . . and that Microsoft does not use the market power of Windows 95 to distort market choice in its favor"). The Government wants to employ a network-externalities argument, but because it is too cautious to develop it fully, the Government falls back on doctrinal arguments. See *id.* (stating that it is free market principles that should decide "battle" between Microsoft and its competitors).

75. See *United States v. Microsoft Corp.*, 1998-2 Trade Cas. (CCH) ¶ 72,621 (D.D.C. Sept. 14, 1998) (No. 98-1232) ("Microsoft entered into a series of anti-competitive agreements with customers and competitors to restrict the use of Java and to substitute the use of Microsoft's version of Java, known as 'J/Direct.'")

76. See Amended Complaint, *supra* note 2, ¶¶ 52-54 (alleging that Microsoft has embarked on course of conduct to disrupt and impair standardized programming environment created by Sun's JAVA technology).

77. See *id.* ¶¶ 53-54 (asserting that Microsoft's actions threaten Java's unique "cross-platform programming environment"). Sun's Amended Complaint described Java's threat to Microsoft in the following manner:

In contrast to the open systems cross-platform programming environment uniquely created by Sun's JAVA Technology, the programming environments established by defendant Microsoft's various operating systems are platform-dependent, such that programs created for it's Win32 and other programming environments will operate only on platforms running defendant Microsoft's operating systems, and will not run on any other systems platform. In a further effort to maintain control and dominance over the programming environment created by its operating systems, defendant Microsoft refuses to license other systems manufacturers, such as Sun and others, to use Microsoft's technology or intellectual property rights to produce operating systems that implement the Win32 programming environment.

Id. ¶ 50. Sun then charged Microsoft with extending their dominant market share position in desktop systems by causing independent software developers either to "create versions of their programs only for the Win32 programming environment, or to so favor the development of Win32-based programs over all other programming environments as to render alternative programming environments for the desktop commercially unattractive for program development." *Id.* ¶ 51.

platform independence.⁷⁸ If Java can be fragmentized into different versions, then the largest market would be Windows users and Microsoft's power would remain intact.⁷⁹

The contentions of the Government and Sun each make reference to the same scenario. Both draw upon the Java-based scenario under which the Windows monopoly is replaced by a browser platform. In the contempt proceeding, the Government implied (rather than asserted) that once Microsoft Explorer became the dominant browser, Microsoft would attempt to undermine the Java potential for engendering a new world of platform-independent computer applications. In its new antitrust case, the Government has explicitly charged that Microsoft's attempts to increase its share of the browser market are motivated by fear of this Java-based scenario.⁸⁰ Although Sun's case does not raise antitrust issues, Sun does incorporate market-structure issues into its lawsuit.⁸¹ Indeed, Sun articulated the missing part of the Government's real case. Sun, like the

78. See *id.* ¶ 54 (alleging that Microsoft intends to disrupt JAVA's cross-platform compatibility). Sun alleged that Microsoft "has embarked on a course of conduct to disrupt and impair" the standardized approach of Java technology "by deliberately implementing Sun's JAVA Technology in products Microsoft distributes in a manner that breaks the cross-platform compatibility of the JAVA programming environment" *Id.*

79. See *id.* ¶ 86. Sun also expressed concern that continual modifications to the Java technology distributed by Microsoft through its Software Development Kit for Java (SDKJ) and Internet Explorer 4.0 have caused and will continue to cause widespread and irreparable harm to:

a) Persons using Java Compatible browsers other than Internet Explorer 4.0, who are harmed because some programs written by developers using Microsoft's SDKJ are not fully functional on their browsers; b) Software developers using SDKJ, who are harmed because, unbeknownst to them, some of the programs they develop are not fully functional on browsers other than Internet Explorer 4.0; c) Browser developers other than Microsoft, who are harmed because some programs written to run by developers who use Microsoft's SDKJ developer's tool kit are not fully functional on browsers other than Internet Explorer 4.0, and this failure is and will likely continue to be mistakenly attributed to such browsers or to Sun's Java Technology; d) Enterprises developing programs for the Java programming environment, who are harmed because the promise of cross-platform compatibility on which they rely is being impaired as a result of Microsoft's efforts to fragment the Java programming environment; and e) Sun, which is harmed because the cross-platform compatibility created by the Java Technology has been and will continue to be impaired, and Sun, not Microsoft is and will continue to be wrongfully identified by developers as the source of their problems.

Id. ¶ 96.

80. See Complaint, *supra* note 1, ¶¶ 8, 68, 73, 107, 122 (containing allegations about threat of competing internet browsers to Microsoft's Windows dominance).

81. See Amended Complaint, *supra* note 2, ¶¶ 48-54 (containing allegations of Microsoft's market dominance). Sun pointed to Microsoft's "dominant market share position" in distributing MS-DOS, Windows 3.1 and Windows 95 operating systems. See *id.* ¶ 49. Sun further alleged that in 1996, over 85% of the desktop systems throughout the world utilized and were dependent on Microsoft's operating systems. See *id.*

Government, brought up the threat to Microsoft's operating systems dominance posed by Java, but in a more articulate and precise manner, Sun explicitly connected that threat to an alleged reaction by Microsoft.⁸² Sun alleged that Microsoft was modifying Java to undermine Java's promise of interoperability, thereby restoring Microsoft's Windows-based dominance.⁸³

The following sections explore the issues surrounding Java's threat to Microsoft. These sections argue that the Government's tying (and related monopolization and attempted monopolization) case is weak, both on the doctrinal terms in which the Government has chosen to state it and on the assumption that Microsoft is seeking to employ an expanding share of the browser market to block the advance of superior technology—a contention that the Government asserts by implication. As a part of the antitrust discussion, this Article examines the contentions of Sun Microsystems which could contribute to that antitrust analysis.

C. *The Technology*

1. *Operating Systems*

All computer systems employ an operating system that acts as a link between application programs and the machine.⁸⁴ The operating system links application programs to the machine while coordinating their operation with an array of functions, such as printing or copying, that are nor-

82. *See id.* ¶ 52 ("Microsoft seeks to extend the dominance it currently enjoys over desktop operating systems and the programming environments they create to browser programs and other applications through its manufacture and distribution of Internet Explorer 4.0 and its Software Development Kit for JAVA ('SDKJ')"). Sun alleged that Microsoft's reaction to Java's threat to its operating systems dominance was its manufacture and distribution of Internet Explorer 4.0 and its SDKJ. *See id.* ¶¶ 48-54 (containing allegations about Microsoft's strategy concerning Java).

83. *See id.* ¶¶ 75-77 (detailing Microsoft's alterations of Sun's Java's Development Kit (JDK) 1.1). Among other charges, Sun asserted that SDKJ both added and deleted elements of Sun's JDK 1.1 and failed to contain or support critical portions of Sun's JDK 1.1 upgrade to the Java technology. *See id.* In particular, Sun alleged that Microsoft altered and modified Sun's set of Java application programming interfaces (APIs) contained in JDK 1.1 by deleting the Java Native Method Interface (JNI). *See id.* ¶ 79. It also allegedly eliminated the package of supplemental class libraries called "Remote Method Invocation" (RMI), and did not otherwise make RMI separately available through alternative channels of distribution as required by the TLD Agreement. *See id.*

84. *See* STEVEN MANES & PAUL ANDREWS, GATES: HOW MICROSOFT'S MOGUL REINVENTED AN INDUSTRY—AND MADE HIMSELF THE RICHEST MAN IN AMERICA 107-08 (1994) (discussing function of computer operating system). An operating system coordinates the different demands that software places on the hard drive. *See id.* A computer without an operating system is unable to do such simple tasks as store files on a disk, or send information to a printer, without specific directions from a programmer. *See id.* The operating system acts as a built-in programmer and performs these functions on demand. *See id.*

mally desired by users, but which are not contained within applications programs.⁸⁵

Since the early 1980s, Microsoft has produced the dominant operating system for IBM-compatible personal computers.⁸⁶ Because IBM-compatible personal computers dominate the personal computer market, Microsoft's operating systems have dominated the market for personal computer operating systems. The Microsoft operating system began as a 16-bit MS-DOS designed for the original IBM personal computer and evolved through several stages.⁸⁷ The introduction of the Windows graphical user interface in the early 1990s met a major user need and solidified the dominance of the Microsoft platform, which then consisted of its DOS operating system and a Windows overlay.⁸⁸ A further transformation of the Microsoft platform occurred in 1995 when Windows 95 was introduced.⁸⁹ Windows 95 is a transitional operating system that (while backwardly compatible with 16-bit applications) moves personal computer use to a 32-bit platform. Currently, almost all newly manufactured personal computers equipped with an Intel microprocessor and destined for the consumer market are equipped at the factory with Microsoft's Windows 98 operating system.⁹⁰ Most personal computers manufactured between Au-

85. See Memorandum, *supra* note 12, at 5 (explaining interaction between computer operating system and other computer devices). As stated in the Government's Memorandum in the contempt proceeding, "the operating system is the software that controls the operation of the PC and manages the interaction between the computer's memory and attached hardware devices such as the keyboard, display screen, disk drives, and printer." *Id.*

86. See Gifford, *Microsoft Corporation*, *supra* note 33, at 624-28 (describing Microsoft's dominance of PC operating systems since early 1980s); see also LISA BUCKI, PCS: 6 IN 1 8 (1997) ("Every personal computer comes with the operating system software. Microsoft Windows 95 ships on most new personal computers today . . ."); JAMES WALLACE, OVERDRIVE: BILL GATES AND THE RACE TO CONTROL CYBERSPACE 8 (1997) ("First with DOS and then with Windows, Microsoft had become dominant because it controlled the operating system for most of the world's personal computers.").

87. See generally Gifford, *Microsoft Corporation*, *supra* note 33, at 624-28 (commenting on Microsoft's development of its operating system through six successive stages).

88. See MANES & ANDREWS, *supra* note 84, at 6 (detailing development of graphical user interface). One commentator has noted that the graphical user interface concept "was invented neither by Bill Gates nor by Microsoft." *Id.* Instead, Windows was a descendant of "earlier graphical user interfaces, most notably the experimental versions from Xerox's Palo Alto Research Center . . . and the commercial version from Apple Computer's Macintosh." *Id.*

89. See ED TILEY, WINDOWS 95 UNLEASHED 4 (1998) (stating that Windows 95 is full-blown operating system, not just layer over DOS).

90. See Russ Mitchell & Marianne Lavelle, *Lots of Legal Bugs: Microsoft Has Been Rushing to Roll Out Windows 98*, U.S. NEWS & WORLD REP., Apr. 27, 1998, at 54, 54 (noting that by July 1998, all new computers sold by Compaq, Gateway 2000, Dell and others will have Windows 98 installed on hard drive).

gust 1995 and May 1998 were equipped with Windows 95.⁹¹ Most older machines are equipped with a version of MS-DOS and Windows 3.1 graphical interface overlay, both products of Microsoft.⁹²

Operating systems coordinate the functions performed by the various hardware components of a personal computer: the keyboard, the screen, the random access memory (RAM), the functioning of the hard drive, the printer, etc.⁹³ They also generally contain several utility programs, such as programs for formatting data carriers.⁹⁴ Over time—sometimes in response to the growth of memory provided by the hardware—operating systems have absorbed new functions.⁹⁵ A comprehensive file management system is one example. As late as 1995, one author commented that although “[a]ll operating systems must . . . provide several utility programs supporting the general use of the PC system,” nonetheless “an extremely powerful and user-friendly file manager, which would support common file management tasks, such as copying, moving, or deleting files, is more difficult to find.”⁹⁶ Early personal computer operating systems lacked such file management capabilities. Microsoft operating systems, for example, began to carry such a file manager (called DOSSHELL) in MS-DOS version 4.0.⁹⁷ Printing is another function that has sometimes existed independently. Printing is now accomplished through coordinated action between applications software and the operating system, at least those carrying Microsoft and Macintosh operating systems.⁹⁸ In UNIX systems,

91. See generally BUCKI, *supra* note 86, at 14 (stating that, in 1997, most consumers purchase computers run by Windows 95 operating system); WALLACE, *supra* note 86, at 279 (discussing Windows 95 sales).

92. See generally ULRICH SCHUELLER & HANSS-GEORG VEDDELER, *UPGRADING & MAINTAINING YOUR PC 195-200* (3d ed. 1995) (providing brief history of Windows).

93. See BUCKI, *supra* note 86, at 8 (explaining that purpose of operating system software is to instruct computer how to start up, communicate with all its devices and work with other software).

94. See SCHUELLER & VEDDELER, *supra* note 92, at 183 (discussing operating system programs); see also DAVIS, *supra* note 12, at 2-3 (illustrating functions performed by operating system). One commentator explained that:

The operating system's routines perform key support functions such as communicating with peripheral devices and accepting and carrying out user commands (load a program, copy a file, create a directory . . .). These seemingly simple tasks are, in reality, deceptively complex. They are also common to most applications, and it makes little sense to duplicate them in each and every application program.

Id.

95. See RONALD E. ANDERSON & DAVID R. SULLIVAN, *WORLD OF COMPUTING* 114 (1988) (describing evolution of operating systems); SCHUELLER & VEDDELER, *supra* note 92, at 183 (observing development in personal computer operating systems as result of constantly evolving hardware technology).

96. *Id.*

97. See *Operating Instructions; 1,001 Tips: Operating Environments*, PC COMPUTING, Aug. 1994, at 108, 133 (discussing DOSSHELL file-management utility that Microsoft introduced in Version 4.0).

98. See Memorandum, *supra* note 12, at 5 (explaining interaction between computer operating system and other computer functions, including printing).

however, printing is an independent function.⁹⁹ Perhaps the most dramatic (and visible) function absorbed by an operating system has been the graphical user interface.¹⁰⁰ Early Microsoft operating systems did not have a graphical user interface. When Microsoft introduced its Windows graphical user interface in the early 1990s, it first appeared as an overlay on the operating system (as in Windows 3.1).¹⁰¹ Later—with Windows 95—it was fully integrated into the operating system.¹⁰²

2. *The Internet and the World Wide Web*

The internet is a vast network created by the interconnection of servers with computer users through telephone lines.¹⁰³ The World Wide Web is that part of the internet that uses Hypertext Transport Protocol (HTTP) to link documents across the internet.¹⁰⁴ It consists of (1) servers that provide “pages” and other material and (2) browsers employed by viewers to “read” the material provided by servers.¹⁰⁵ Therefore, the software written for servers must be compatible with the capabilities of browsers; otherwise, the material dispensed by the servers would be unreadable. Currently, most material deployed on web servers is written in HTML language, often on a UNIX operating system.¹⁰⁶ HTML is in the process of being replaced by DHTML, and HTML programming is being

99. See generally KEVIN REICHARD & ERIC F. JOHNSON, *TEACH YOURSELF . . . UNIX 77* (3d ed. 1995) (noting that UNIX operates as hierarchical file system, with every function, including printing, located in separate file).

100. See PHILIP E. MARGOLIS, *PERSONAL COMPUTER DICTIONARY* 217 (2d ed. 1996) (defining graphical user interface (GUI)). A GUI takes advantage of the computer's graphics capabilities to make the program easier to use. See *id.* Microsoft Windows' GUI features the following basic components: (1) pointer—a small symbol that appears on the screen that allows the user to select various options; (2) pointing device—the physical device that enables the user to direct the pointer across the screen; (3) icons—small pictures that represent commands that are easier to identify for the user; (4) desktop—the area on the computer screen where the icons are grouped; (5) windows—the ability to divide the computer screen into different areas; and (6) menus—the ability to execute commands from selections on a menu. See *id.* at 217-18.

101. See MANES & ANDREWS, *supra* note 84, at 6-8 (discussing evolution of graphical interface).

102. See MARK MINASI, *THE EXPERT GUIDE TO WINDOWS 95* 2-7 (1996) (noting multiple improvements in Windows 95's GUI as compared to older Windows' GUI).

103. See John Markoff, *Microsoft Seems Near Deal to Invest in US West Cable TV*, N.Y. TIMES, Nov. 5, 1997, at D1 [hereinafter Markoff, *Microsoft Seems Near Deal*] (indicating that most households connect to internet via conventional telephone lines).

104. See *How the WWW Is Put Together*, BYTE, Aug. 1995, at 138, 138 (discussing constitution of World Wide Web and indicating that HTTP is most important protocol used by Web).

105. See *id.* (explaining interaction of server and browser).

106. See Tom Yager, *NT and the Net*, BYTE, July 1996, at 133, 133 (noting that UNIX has thus far dominated Web development).

increasingly supplemented with Java additions.¹⁰⁷ So far, Java has been concentrated in the production of “applets,” dynamic portions of web pages that can interact with a viewer through the provision of small spreadsheet programs, animation or video clips.¹⁰⁸ Standardization in internet communication is overseen by the World Wide Web Consortium (“W3C”), which administers the content of a number of protocols governing internet communication.¹⁰⁹

3. *Browsers*

Browsers are a relatively new invention.¹¹⁰ The first browser with a convenient graphical user interface, the Mosaic, was developed in 1993 at the National Center for Supercomputer Applications (NCSA) at the University of Illinois.¹¹¹ Mosaic was followed by the Netscape Navigator, which, for a substantial period, was given away free-of-charge to capture

107. See generally Rick Dobson, *Dynamic HTML and Scriptlets Add Life*, BYTE, Jan. 1998, at 79, 79 (providing insight into dynamic hypertext markup language). Although initially Microsoft and Netscape were implementing DHTML differently, the WC3 is expected to issue a clarifying standard, ending the disparity. See *id.* (noting that Microsoft and Netscape have committed to “interoperability pledge,” stating that their browsers will comply with WC3 specifications). XML, extensible markup language, may ultimately displace DHTML. See Scott Mace et al., *Weaving a Better Web*, BYTE, Mar. 1998, at 58, 62-65 (noting momentum behind XML and predicting that XML will become vehicle for publishing SGML-based information on Web).

108. See Dobson, *supra* note 107, at 79 (discussing how applets include properties that can enliven web pages); see also Stephen J. Vaughan-Nichols & Rachel Schmitter, *How the Web Will Change Computing*, BYTE, Jan. 1996, at 24, 24 (describing how applets can include data-entry forms, spreadsheets, animations and graphics).

109. See, e.g., David Lytel, *Nonprofit Parents, Corporate Kids*, UPSIDE, Mar. 1998, at 66, 66 (discussing need for internet oversight); see also Michael Moeller, *Standards Could Put Bite on Unwanted Web Data; Consortium Creating Hypertext Extensions For Filtering Content*, PC Wk., Sept. 11, 1995, at 1, 1 (noting that consortium is developing extensions to hypertext that will allow user opportunity to screen, rate and filter internet content).

110. See Peter Elstrom et al., *Silicon Valley: Beyond the Valley*, BUS. WK., Aug. 25, 1997, at 138, 138 (describing history of first web browser). Marc Andreessen, Vice-President of Netscape, while a student at the University of Illinois in the early 1990s, helped to develop the Mosaic, the first convenient browser with easy-to-use graphical format. See *id.* Mosaic was released to the public in April 1993. See Steve Hamm, *The Education of Mark Andreessen*, BUS. WK., Apr. 13, 1998, at 84, 86. After a dispute with the University of Illinois over the rights to Mosaic, Andreessen helped to form Netscape. See Elstrom et al., *supra*, at 138. Netscape then marketed the first commercial browser in 1994, followed by several improved versions thereafter. See Robert D. Hof, *Netspeed at Netscape*, BUS. WK., Feb. 10, 1997, at 79, 81 [hereinafter Hof, *Netspeed*]. Spyglass, Inc. won the right to market Mosaic in 1994. See Peter Elstrom & Kathy Rebello, *A Big Bet on Minibrowsers*, BUS. WK., Oct. 28, 1996, at 56, 56.

111. See Hamm, *supra* note 110, at 86 (providing time line of Mosaic’s development).

market share and stimulate the market for servers.¹¹² By 1995, Netscape commanded over eighty percent of the browser market.¹¹³ In that year, Netscape incorporated Java technology into its Navigator 2.0.¹¹⁴ Since that time, websites have been able to use Java applets to enhance their capabilities and overall attractiveness.¹¹⁵ Netscape's success in the browser market engendered a vast production of material for servers and encouraged entry by others into the browser market. Microsoft shipped its first browser in 1994, released its improved Explorer 2.0 in mid-1995 and was giving it away by year's end.¹¹⁶ The Explorer has been increasing its market share since 1995.¹¹⁷ By 1997, both Netscape and Microsoft had issued highly sophisticated versions of their respective browsers, the Netscape Communicator and the Microsoft Internet Explorer 4.0.¹¹⁸ Sun Microsystems also produces a browser, which it calls "HotJava."¹¹⁹ The current versions of these browsers are equipped with Java capability. As a result of Microsoft's increasing penetration of the browser market, Netscape announced in early 1998 that it would soon resume its earlier prac-

112. See Ben Smith, *Internet With Style*, BYTE, Jan. 1995, at 197, 198 (discussing Netscape's business practice); see also Ellis Booker, *Netscape Keeps on Rolling*, COMPUTERWORLD, Mar. 6, 1995, at 32, 32 (referring to "Netscape's most canny move: Leveraging a free-for-the-taking piece of client software as a means of selling a server product"); Peter H. Lewis, *Business Technology: Netscape Knows Fame and Aspires to Fortune*, N.Y. TIMES, Mar. 1, 1995, at D1 (revealing strategy behind Netscape's practice of giving away bulk of its software for free).

113. See generally Heather Green, *Has Netscape Hit the "Innovation Ceiling"?*, BUS. WK., Jan. 19, 1998, at 69, 69 (indicating that in web browser market, Netscape's share had fallen from high of 85% to 60% by January 1998).

114. See Dragan & Seltzer, *supra* note 51, at 100 (stating that in 1995, Netscape incorporated Java support in Navigator 2.0).

115. See *id.* (stating that most widespread use of Java is on Web "where developers have employed it for creating applets—interactive page elements or basic applications that you view within a browser"). Java-enabled browsers include HotJava, Microsoft Internet Explorer, Netscape Communicator and Netscape Navigator. See *id.*

116. See Kathy Rebello et al., *Inside Microsoft*, BUS. WK., July 15, 1996, at 56, 56 (explaining Microsoft's browser development). Spyglass, Inc. won the right to Mosaic, and licensed the Mosaic technology to Microsoft when the latter decided to enter the browser market. See Elstrom & Rebello, *supra* note 110, at 56 (noting Spyglass' attempt to get back into browser market). See generally *Senate Hearing*, *supra* note 10 (discussing testimony of Bill Gates regarding licensing arrangement with Spyglass).

117. See William Echikson, *Bill is a Target in Europe, Too*, BUS. WK. INT'L EDITION, Feb. 23, 1998, at 14, 14 (stating that market share of Microsoft's Internet Explorer was 40% worldwide). According to the Justice Department, Microsoft held approximately a 50% share of the browser market in May 1998. See Complaint, *supra* note 1, ¶¶ 64, 126.

118. See *Downloads and Software*, PC/COMPUTING, Aug. 1998, at 138, 138 (stating that Microsoft Internet Explorer 4.01 and Netscape Communicator 4.05 are "so complete they make older versions pale in comparison").

119. See Andrew Singleton, *Wired on the Web*, BYTE, Jan. 1996, at 77, 77 (discussing development of HotJava).

tice of giving its browser away.¹²⁰ In addition, Netscape announced that it would make the source code for its browser generally available.¹²¹

The practice of giving browsers away is neither economically irrational nor predatory behavior.¹²² Both Netscape and Microsoft apparently regard the browser "market" as essentially a means for advancing their position in the server market.¹²³ Both companies market packages of server/browser software for intranet use (*i.e.*, for use internally within the networks of private business enterprises).¹²⁴ They also believe that they can increase the marketability of their server software as they increase their share of the browser market.¹²⁵

4. *Java*

Java is a programming language with characteristics that fulfill, or have the potential for fulfilling, a number of current or perceived needs in programming and in internet communication.¹²⁶ Java possesses capabilities that are lacking in C++, which up until now has been the most widely employed programming language.¹²⁷ Indeed, Java grew out of C++ and embodies much of the C++ language.¹²⁸ Java did, however, discard certain C++ elements that heightened error vulnerability.¹²⁹ Currently, one

120. See Don Clark, *Netscape to Share Browser Code*, WALL ST. J., Jan. 23, 1998, at B6 (indicating that Microsoft's successful assault on web browser market will cause Netscape to begin giving away its browser software again).

121. See *id.* (discussing Netscape's plan to share its programming code, which will allow other companies to enhance it).

122. But see Lemley, *supra* note 69, at 1075 (describing behavior of both Netscape and Microsoft in giving away browsers as "competitive predation"). Lemley has suggested that there is reason to be concerned about such behavior only to the extent that one firm is better able to survive the period of below-cost pricing. See *id.* at 1075-76. The Government has questioned Microsoft's behavior in giving the browser away, but it does not refer to a similar practice by Netscape. See Complaint, *supra* note 1, ¶¶ 120, 121 (containing allegations about giving away browser).

123. For a discussion of Netscape's business practices, see *supra* note 112 and accompanying text.

124. See Sunny Baker & Kim Baker, *The 1998 Strategist's Guide to Software*, BUS. STRATEGY, Jan.-Feb. 1998, at 39, 43 (discussing Netscape's foray into "superserver domain" and Microsoft's Internet Information Server bundled with its NT Server Software).

125. See Robert D. Hof, *An Instant Success! So Where's the Payoff?*, BUS. WK., June 15, 1998, at 126, 126 [hereinafter Hof, *Instant Success*] (noting that Netscape's practice of giving away web browser is means to build demand for its lucrative server software).

126. For a discussion of the uses and unique qualities of Java, see *infra* notes 127-43 and accompanying text.

127. See Ralph Spindell, *New Development Tools Keep Java Percolating*, ELECTRONIC DESIGN, July 20, 1998, at 35, 35 (explaining how Java could replace C++ or Virtual Basic as programming language of choice for professionals).

128. See DAVID FLANAGAN, *JAVA IN A NUTSHELL* 14 (2d ed. 1997) (stating that Java borrows terminology and syntax from C++).

129. See Richard V. Dragan, *Java Tools Get Real*, PC MAG., Jan. 7, 1997, at 181, 181 [hereinafter Dragan, *Java Tools*] ("Java's safer language features, such as built-

of Java's important uses is that of facilitating the construction of small computer programs, generally referred to as applets, that enable a web page to interact with a user's computer, providing enhancement to the web page in the form of animation, video clips or other dynamic material.¹³⁰ In addition to superior capabilities as a language, Java also has the potential for interoperability, allowing a program written in Java to run on any platform, including Windows/Intel, OS/2, Macintosh, Linux and UNIX.¹³¹ Java is independent of the processor architecture and of the operating system in the host computer.¹³²

Java is platform-independent because, unlike other software, standard Java is not compiled into machine code. Rather than compiling source code into object code designed for a particular platform as compilers generally do, Java compilers transform Java source code into "Java byte code."¹³³ It is these Java byte codes that are designed to be executed on any system that has a Java byte code interpreter (or Java virtual machine).¹³⁴ These virtual machines, which are designed specifically for the various platforms (Windows, Macintosh, OS/2, UNIX or other), perform the bridging function connecting the Java program to the operating system and thence to the hardware.¹³⁵

Thus, programs written in Java go through an extra step. Because they are compiled into Java byte code (rather than into object code), they must interface with the interpreter (or virtual machine) in the host machine prior to execution.¹³⁶ This extra step slows the execution of Java programs.¹³⁷ The slowness of Java programs has been repeatedly cited in

in garbage collection and the elimination of pointers, offer an elegant, up-to-date interpretation of C++ without that language's worst traps.").

130. See PAUL MCFEDRIES, *THE COMPLETE IDIOT'S GUIDE TO CREATING AN HTML 4 WEB PAGE* 194 (3d ed. 1998) [hereinafter MCFEDRIES, *IDIOT'S GUIDE*] (exploring uses and unique qualities of Java).

131. See FLANAGAN, *supra* note 128, at 5 (explaining that writing program in Java allows user to run it on all platforms).

132. See *id.* at 4 (stating that Java applications can run on any system because they are compiled to "an architecture-neutral byte-code format").

133. See Tom R. Halfhill, *Java Chips Boost Applet Speed*, *BYTE*, Apr. 1996, at 25, 25 [hereinafter Halfhill, *Java Chips*] (explaining that Java compiler translates source code into condensed format known as byte code).

134. See *id.* (noting that Java byte code can run on any computer that has Java virtual machine).

135. See generally Amjad Umar, *Application Development: Inside Web App Functioning*, *INFO. WK.*, Sept. 29, 1997, available in 1997 WL 14148086 (explaining that Java virtual machines interpret Java byte codes and run them on hardware).

136. See Kelvin Nilsen, *Adding Real-Time Capabilities to Java*, *COMM. OF THE ACM*, June 1998, at 49, 49 (noting that every Java program is applied to virtual machine's byte code analyzer before it is executed).

137. See Elaine X. Elliot, *New CPUs Make Java Faster, But Less Open: Sun Offers Low-Cost Dedicated Chips to Improve Performance*, *COMPUTER SHOPPER*, June 1996, at 66, 66 (noting that virtual machine interpretation reduces speed of Java applications).

computer magazines.¹³⁸ A related problematic aspect of Java—also connected with its interoperability characteristic—lies in “bugs,” or failures encountered when Java input fails to mesh well with one or more user platforms.¹³⁹ Java, of course, is at an early stage in its development, and these problems may well be resolved in the next few years as its development proceeds.¹⁴⁰

One method for speeding the operation of Java is a “just-in-time” (JIT) Java compiler rather than a Java interpreter.¹⁴¹ A JIT compiler is faster than an interpreter because the JIT compiler keeps native machine code sequences in its memory and uses them when the sequence reappears.¹⁴² The Microsoft Internet Explorer 4.0, for example, employs a JIT Java compiler for its Java virtual machine, thereby speeding the transmission of Java-language programs.¹⁴³

5. *Java Rivals*

Although Java possesses highly desirable capabilities, it is not without rivals. A recently developed rival to Java is IBM’s universal virtual machine, which is designed to enable programs written in Smalltalk, Basic and Java to be translated into six platforms: Windows NT, Windows 95, Hewlett-Packard’s HP-UX, Sun’s Solaris, and IBM’s AIX and OS/2.¹⁴⁴ Another Java rival is Microsoft’s ActiveX.

138. See, e.g., Sharon Gaudin, *Java’s Speed, Cross-Platform Issues Targeted*, COMPUTERWORLD, Apr. 7, 1997, at 1, 1 [hereinafter Gaudin, *Java’s Speed*] (stating that Java is too slow). In fact, Sun has been developing a special picoJava chip as a means of enhancing Java performance. See Peter Wayner, *Sun Gambles on Java Chips*, BYTE, Nov. 1996, at 79, 79. Although picoJava chips have been said to outperform Pentiums with JIT Java compilers, this route to enhancing Java speed is dependent upon the employment of dedicated hardware by users. See *id.* at 79-80 (“Sun says the picoJava chips could be five times faster than a Pentium with a JIT compiler Nevertheless, the actual performance improvement you get will depend on whether the Java program is heavy on computation and light on object juggling.”). In addition to their role in dedicated Java computers, Java processors could also be used as high-speed Java coprocessors in general purpose computers. See Halfhill, *Java Chips*, *supra* note 133, at 25.

139. See Gaudin, *Java’s Speed*, *supra* note 138, at 1 (“[T]here are glitches in its much-touted cross-platform compatibility.”).

140. See Tom R. Halfhill, *Java Gets Down to Business*, BYTE, Oct. 1997, at 87, 88 [hereinafter Halfhill, *Java Gets Down*] (noting that computer coders are confident that Java development will get smoother as tools keep getting better).

141. See Wayner, *supra* note 138, at 79 (noting that just-in-time (JIT) compilers can run Java code even faster than interpreters).

142. See Stephan Somogyi & Jason Snell, *Java Finally Delivers*, MACWORLD, May 1, 1998, at 81, 81 (stating that JIT compilers are faster than typical virtual machines because typical virtual machines do not have memory capacity that compilers do).

143. See Halfhill & Gallant, *supra* note 60, at 70 (explaining how Microsoft Internet Explorer 4.0 performs faster with JIT compiler).

144. See Sharon Gaudin, *IBM Plan Builds on Java*, COMPUTERWORLD, June 23, 1997, at 61, 62 [hereinafter Gaudin, *IBM Plan*] (discussing benefits of IBM’s universal virtual machine).

Introduced in March 1996, ActiveX is not a computer language like Java, but rather a framework embodying a series of application programming interfaces (APIs) that facilitate interaction between the host computer and internet sites in new or existing object-link-embedding (OLE) code.¹⁴⁵ ActiveX grew out of Microsoft's development work on compound documents, work that spawned a broader infrastructure technology concerned with enabling one unit of software to interact with another.¹⁴⁶ Like Java, ActiveX enables programmers to embed small programs (analogous to the Java applets) into websites.¹⁴⁷ ActiveX is designed for use in web browsers, but is geared specifically for the Windows environment.¹⁴⁸ Microsoft takes the view that ActiveX and Java are complementary tools that can work together.¹⁴⁹ In fact, the Java virtual machine contained in the Internet Explorer 4.0 is encased in an ActiveX interface.¹⁵⁰

III. PATH DEPENDENCY, NETWORK EFFECTS AND THE DEVELOPMENT OF COMPUTER SOFTWARE

A. *The Economics*

Network externalities and path dependency theory help to explain the dominance of Microsoft in the market for personal computer operating systems. As discussed below, these concepts have relevance to the development of the Java platform paradigm, but they must be applied with caution.¹⁵¹

145. See Kim S. Nash, *Active X Spells Java Alternative*, COMPUTERWORLD, Mar. 18, 1996, at 14, 14 (describing development of ActiveX).

146. See David Chappell & David S. Linthicum, *ActiveX Demystified*, BYTE, Sept. 1997, at 56, 58 (explaining origin of ActiveX and how it allows interaction between different pieces of software). ActiveX is derived from Microsoft's Component Object Model (COM). See *id.* at 56 (describing ActiveX's relationship to Microsoft's COM. COM grew out of Microsoft's object-link-embedding (OLE) technology, a technology designed to create compound documents. See *id.* at 58. COM was contained in the second release of OLE, OLE 2. COM, however, possessed broader capabilities than those needed for compound documents. See *id.* COM embodied technology facilitating one piece of software to interact with another. See *id.* Although COM is employed in a whole array of Microsoft products, it is a subset of COM technologies that constitutes ActiveX. See *id.* at 56 (stating that ActiveX is built from COM technologies).

147. See Colin Savage, *Sound Off: Java vs. ActiveX: You Say You Want a Revolution*, COMPUTERWORLD, May 12, 1997, at 80, 80 (describing beneficial aspects of ActiveX).

148. See *id.* ("ActiveX is useful only for extending Windows applications.").

149. See *Apple's Future Tied to Java*, NEWSBYTES, May 28, 1996, available in 1996 WL 10475865 ("The Microsoft model is to couple Java and ActiveX together. According to Microsoft, Java will work better wrapped in the Microsoft ActiveX system.").

150. See Richard V. Dragan, *What's the Code?*, COMPUTER SHOPPER, Apr. 1, 1998, at 522, 522 [hereinafter Dragan, *Code*] (explaining how Java and ActiveX work together on Internet Explorer).

151. For a discussion of network externalities and path dependency theory, see *infra* notes 152-60 and accompanying text.

Both network externalities and path dependency theory apply to many products whose value increases as a result of the growth of a network in which the product participates.¹⁵² The paradigmatic example is the telephone. One telephone is worthless, but its value increases as more telephones are added to the telephone network. Personal computer operating systems, like telephones, tend to grow in value as they increase in number. Thus, the larger is the number of users of an operating system (*i.e.*, the greater the installed base), the greater is the incentive of software applications developers to write programs for that system. Developers may write programs for other operating systems as well, but the market represented by the dominant operating system is their first priority because it is the largest. As a result, the dominant operating system will have more software available for use than other operating systems. This fact, in turn, increases the attractiveness of the dominant operating system to new computer users who tend to select that system in preference to rival systems.¹⁵³ This is the phenomenon known as network externalities.¹⁵⁴ The value of the operating system increases, not because of any inherent physical attribute, but merely because of its relationships with other products that can be used with it. As more buyers select the now-dominant operating system, that operating system grows still further, which, in turn, generates even stronger incentives in software developers to write for the dominant system.¹⁵⁵ The growth of the operating system and the growth of applications software programs exert mutually reinforcing effects spiraling in their intensity over time. As the operating system becomes increasingly attractive as a result of the interaction of its growth with the software applications industry, it becomes ever more difficult for a rival operating system to gain widespread acceptance. Theorists conceptualize this process as path dependency, the ultimate result determined by prior events—the prior events here being the positive feedback provided by the mutually

152. See W. Brian Arthur, *Competing Technologies, Increasing Returns, and Lock-In by Historical Events*, 99 *ECON. J.* 116, 122 (1989) (discussing how path efficiency is provable only in constant and diminishing returns scenarios); Michael L. Katz & Carl Shapiro, *Systems Competition and Network Effects*, 8 *J. ECON. PERSP.* 93, 94 (1994) [hereinafter Katz & Shapiro, *Systems Competition*] (explaining how value of membership to users is affected when other users join); S.J. Liebowitz & Stephen E. Margolis, *Network Externality: An Uncommon Tragedy*, 8 *J. ECON. PERSP.* 133, 135 (1994) (defining network externalities).

153. See Katz & Shapiro, *Systems Competition*, *supra* note 152, at 94 (“[T]he availability of software will depend on what other consumers do, which gives rise to positive-feedback effects.”).

154. See Liebowitz & Margolis, *supra* note 152, at 135 (“The circumstance in which the net value of an action (consuming a good, subscribing to telephone service) is affected by the number of agents taking equivalent actions will be called a network effect.”). Network externalities show a “specific kind of network effect in which the equilibrium exhibits unexploited gains from trade regarding network participation.” *Id.*

155. See Katz & Shapiro, *Systems Competition*, *supra* note 152, at 94 (stating that availability of software will depend on what consumers do, and that this gives rise to positive-feedback effects).

reinforcing and interdependent growth in applications software and operating systems. The further along in the process the operating system moves, the more network externalities augment its value and the stronger its lead becomes over rivals.¹⁵⁶

To the extent that the concepts of network externalities and path dependency apply to operating systems, they measure the extent to which a dominant operating system enhances social welfare. The dominance of one system reflects the value placed by users on the network.¹⁵⁷ That dominance is evidence that no alternative operating system possesses such technical superiority as to outweigh the combination of the technical merits of the dominant system plus the value of the network.¹⁵⁸ Conversely, the existence of nondominant systems reflects the values that users place on the technical merits of these smaller systems.¹⁵⁹ The existence of a nondominant system shows that its users have found the combination of the technical merits of that system plus the value of the smaller network to outweigh the value of the larger network and the technology on which the larger network rests.¹⁶⁰

Economies of scale also help to reinforce network effects. An operating system possesses the theoretical attributes of a natural monopoly—it is characterized by high fixed cost incurred in research and development and minimal marginal costs in production and sales.¹⁶¹ Because operating systems, like all software, are replicable at minimal cost, the major cost connected with increased sales is technical backup support and whatever monitoring is necessary to discourage illegitimate copying.¹⁶² Economies of scale, however, do not explain why a particular operating system is dom-

156. See Gifford, *Microsoft Corporation*, *supra* note 33, at 647-48 (explaining movement towards path dependency and its effect on rivals).

157. See Katz & Shapiro, *Systems Competition*, *supra* note 152, at 94 (explaining consumer value of network).

158. See *id.* at 106 (explaining consumer value placed on technology and network size in relation to dominant and rival systems).

159. See *id.* (discussing how consumer heterogeneity and product differentiation limit "tipping" while sustaining multiple networks). Tipping occurs when one system pulls away from its rivals in popularity after gaining an initial edge. See *id.*

160. See *id.* (mentioning that if rival systems have distinct features, two or more systems may be able to survive by catering to consumers who care more about product attributes than network size).

161. See ALFRED D. CHANDLER, JR., *SCALE AND SCOPE: THE DYNAMICS OF INDUSTRIAL CAPITALISM* 17 (1990) (defining economies of scale as "those that result when the increased size of a single operating unit producing or distributing a single product reduces the unit cost of production or distribution"); Douglas Gagax & Kenneth Nowotny, *Competition and the Electric Utility Industry: An Evaluation*, 10 YALE J. ON REG. 63, 67 (1993) (explaining that natural monopoly arises when single firm can produce desired level of output at lower cost than any output combinations for more than one firm).

162. See Baseman et al., *supra* note 15, at 270 (stating that for operating systems "fixed costs are enormous and marginal costs are negligible"); Gifford, *Microsoft Corporation*, *supra* note 33, at 639-40 (discussing scale economies and monitoring costs).

inant, because as volume increases, unit cost decreases, but at a declining rate.¹⁶³ As a result, the potential customer base for operating systems is probably large enough to support a number of them.

Thus, both network effects and scale economies contribute to the support of a dominant operating system. When a system becomes sufficiently dominant as to become a de facto industry standard, positive social welfare benefits are generated. Network effects theory can be used to describe how an industry standard facilitates interaction among users of the system and the development of applications systems at low cost.¹⁶⁴ Scale economies help society to produce the standard system while economizing on resources.¹⁶⁵

B. *Historical Events Contributing to Microsoft's Dominance Over Operating Systems*

Network externalities explains the value of a network in settings in which the use of the network by others enhances each user's welfare.¹⁶⁶ When employed in connection with path dependency theory, it can explain how an inferior technology can achieve a dominant position in a market characterized by network effects.¹⁶⁷ Indeed, an initial lead time—which results from accidents or other trivial events—could confer an advantage on the winning technology.¹⁶⁸ Network effects may enhance that early advantage so that rival technologies never pose a serious threat of displacing the leader. It is also true, however, that the lead held by the dominant technology may be the result, in part, of technical superiority, better marketing or both. In the case of Microsoft, there are reasons to

163. As a fixed cost is allocated among increasing units of output, unit cost necessarily declines. The reduction in unit cost, however, grows smaller as volume increases.

164. See John E. Lopatka & William H. Page, *Posner's Program for the Antitrust Division: A Twenty Five Year Perspective*, 48 SMU L. REV. 1713, 1739 (1995) [hereinafter Lopatka & Page, *Posner's Program*] (explaining how unregulated markets can fail because of incompatible standards); see also Mark A. Lemley & David McGowan, *Legal Implications of Network Economic Effects*, 86 CAL. L. REV. 479, 497 (1998) [hereinafter Lemley & McGowan, *Legal Implications*] (explaining how single standard may promote efficiency in some markets).

165. See CHANDLER, *supra* note 161, at 17 (discussing benefits of economies of scale).

166. See Lopatka & Page, *Posner's Program*, *supra* note 164, at 1738-39 (describing network externalities and explaining how complementary goods become more plentiful and lower in price as number of users for goods increases). See generally Michael L. Katz & Carl Shapiro, *Network Externalities, Competition and Compatibility*, 75 AM. ECON. REV. 424 (1985) [hereinafter Katz & Shapiro, *Network Externalities*] (discussing network effects and implications for antitrust).

167. See Liebowitz & Margolis, *supra* note 152, at 135 (defining network effect as "circumstance in which net value of an action . . . is affected by number of agents taking equivalent actions").

168. See W. BRIAN ARTHUR, *INCREASING RETURNS AND PATH DEPENDENCE IN THE ECONOMY* 19, 24 (1994).

believe that its success is causally connected to the combination of product quality and superior marketing.¹⁶⁹

Microsoft became the producer of the dominant personal computer operating system when it undercut the prices charged by producers of rival operating systems for the original IBM personal computer.¹⁷⁰ When the IBM personal computer was cloned, Microsoft furnished its operating system (MS-DOS) to the clone producers, continuing its dominance.¹⁷¹ During the next decade and a half, Microsoft released several improved versions of MS-DOS.¹⁷² Because each of the upgrades ran applications designed for earlier versions of MS-DOS, Microsoft retained its leadership over operating systems.¹⁷³ After Apple introduced a graphical user interface in its unsuccessful Lisa and later in its successful Macintosh, Microsoft released its Windows graphical user interface in 1990 and 1991 as an overlay for MS-DOS.¹⁷⁴ In Windows, Microsoft again provided an operating

169. See, e.g., Tracy E. Benson, *Beyond Niche Marketing*, INDUSTRY WK., Sept. 17, 1990, at 12, 12-14 (discussing Microsoft's outstanding marketing efforts); Lopatka & Page, *Antitrust Decision Making*, *supra* note 35, at 329 (noting that "story of Microsoft's success is well-known"); Susan Yeaton, *Powerful Presentations for Windows*, BYTE, Jan. 1995, at 183, 183-84 (noting that Microsoft's Powerpoint was best overall product).

170. See BILL GATES, *THE ROAD AHEAD* 49 (1995) (describing pricing of operating systems for original IBM personal computer).

171. See generally Bradley Johnson, *Low End Clones Flourish, Spurred by Recession: Niche Marketing May be the Answer for Top PC Brands*, ADVER. AGE, Apr. 27, 1992, at 37, 37 (discussing IBM's loss of market share to clones because of Intel and Microsoft's marketing to clone-producers). The spread of clones, all using the Microsoft operating system, provided a large installed base attracting software developers to the Microsoft/Intel platform. Apple's decision not to license production of the Macintosh contributed to the smaller base of these machines and thus to their reduced attractiveness to software developers. See Gifford, *Microsoft Corporation*, *supra* note 33, at 650 (explaining how Microsoft's marketing plan provided individuals with higher quality, lower priced operating systems).

172. See Brian Livingston, *MS-DOS 5.0—A Real Improvement?* SYS. INTEGRATION, Aug. 1991, at 40-42 (noting that MS-DOS series of upgrades are useful, but problems still exist); see also Tim Clark, *Early Returns Strong for 2 Microsoft Campaigns*, BUS. MARKETING, May 1993, at 12, 12 (noting that more copies of MS-DOS 6.0 were sold in first two weeks than all of DOS 5.0 and Windows 3.1 combined).

173. See Michael D. Millikin, *Microsoft's Forecast: Hot and Sunny*, DATA COMM., Mar. 1992, at 35, 35 (noting Microsoft's enhancement of products and rating as top vendor).

174. See generally GATES, *supra* note 170, at 53-54 (discussing release of graphical interface); Doug Harper, *The PC Graphical User Interface Arrives: Microsoft Windows Proves There's Life in DOS Yet*, INDUS. DISTRIBUTION, Oct. 1990, at 61, 61 (discussing how graphical interface revitalizes MS-DOS, is user friendly and improves memory management). To develop its graphic user interface, Microsoft employed technology that was licensed in a significant amount from Apple. See MANES & ANDREWS, *supra* note 84, at 291-93.

Microsoft released Windows 3.1 in April 1991. Windows 3.1 was a significant enhancement of Windows 3.0, which had been released some time before. See Baseman et al., *supra* note 15, at 265 (chronicling development of Windows); see also Lopatka & Page, *Antitrust Decision Making*, *supra* note 35, at 322 (discussing development of Windows, which "added to MS-DOS a graphical user interface

device that proved extremely popular. At the time Microsoft offered Windows, a number of rival user interfaces were being offered by competitors, but Windows won out in the market.¹⁷⁵

The success of both MS-DOS and Windows can be described in terms of network externalities and path dependency theory. Initial success fed upon itself, as applications producers wrote applications for the Microsoft products, thus enhancing the value of the underlying MS-DOS and Windows programs and in turn further stimulating applications producers to produce applications compatible with MS-DOS and Windows.

In the 1990s, the personal computer industry faced the transition from 16-bit to 32-bit architecture.¹⁷⁶ In 1992, IBM and Microsoft, which had been jointly developing an operating system (OS/2) to replace MS-DOS, abandoned their joint efforts. IBM then introduced its OS/2 2.0, a 32-bit system, in 1992.¹⁷⁷ Subsequently, Microsoft introduced its own 32-bit operating system, Windows NT.¹⁷⁸ Later, in 1994, IBM released its OS/2 Warp, a revised and improved version of OS/2 and also—like OS/2.0—a 32-bit system.¹⁷⁹ Windows NT was not compatible with the 16-bit applications programs written for MS-DOS and its Windows overlays (such as Windows 3.1).¹⁸⁰ Partially because of this backward incompatibility, Windows NT did not attract many users in the nonbusiness sector. While OS/2 Warp was backwardly compatible, its need for 16 megabytes of RAM discouraged its use at the time of its introduction in the Fall of 1994 because personal computers then did not routinely carry that amount of RAM.¹⁸¹

(GIU) and ability to execute more than one application simultaneously," which is known as multitasking).

175. IBM was then offering its "Topview," a character-based windows and multitasking system. See MANES & ANDREWS, *supra* note 84, at 266-67; see also Albert J. Enzweiler, *The Windows Interface Takes Over*, MGMT. ACCT., Dec. 1995, at 54, 54-55 (noting that desire to work in windows has spread beyond spreadsheets and that all top 45 software vendors have Windows-based ledgers); *Windows Pain: Computer Software*, THE ECONOMIST, May 8, 1993, at 72, 72 (discussing Windows' market domination).

176. See Gifford, *Microsoft Corporation*, *supra* note 33, at 650 (noting that Microsoft is leading transition from 16-bit to 32-bit architecture).

177. See Lopatka & Page, *Antitrust Decision Making*, *supra* note 35, at 322 (discussing development of OS/2 and OS/2 Warp); see also Laurie Hayes & Michael Wofsey, *IBM's OS/2.1 Debut Is Modest Success But Launch of Rival Windows NT Looms*, WALL ST. J., July 2, 1993, at B2 (discussing debut of OS/2 and impending release of Windows NT).

178. See Hayes & Wofsey, *supra* note 177, at B2 (discussing release of Windows NT).

179. See Alan S. Kay, *Where WARP Fits*, DATAMATION, Oct. 15, 1995, at 75, 75-77; Lopatka & Page, *Antitrust Decision Making*, *supra* note 35, at 322 (discussing release of OS/2 Warp as marketing alternative to Windows).

180. See generally, *White Paper—Microsoft Corp.*, COMPUTER RESELLER NEWS, Sept. 29, 1997, available in 1997 WL 13698173 (discussing how Windows NT is not compatible with 16-bit applications).

181. See Kay, *supra* note 179, at 75-77 (discussing OS/2); John McMullen, *OS/2's Second Coming*, DATAMATION, Apr. 15, 1991, at 49, 49-50 (discussing 32-bit ver-

Microsoft successfully managed a transition from the 16-bit system to a 32-bit system with Windows 95. Windows 95 and its successor, Windows 98, run 32-bit programs and are backwardly compatible with the 16-bit applications written for MS-DOS and Windows 3.1.¹⁸² By the Summer of 1995, when Windows 95 was introduced, most computers were being marketed with at least eight megabytes of RAM, an amount that was sufficient for Windows 95.¹⁸³ Thus, contrary to IBM, Microsoft keyed the design of its new product to the capabilities of the prevailing hardware.¹⁸⁴

Unquestionably, network effects made a major contribution to Microsoft's success. They appear, however, to account for only a part of that success. The combination of technology geared to meet the needs and desires of consumers with astute marketing also must be taken into account in any assessment of how Microsoft achieved its present position. Microsoft's present dominance in operating systems does not appear to be explainable as the result of one or more trivial events, as in the stereotype often referred to in the network externalities literature.¹⁸⁵

IV. THE FUTURE OF THE SOFTWARE INDUSTRY: THE GOVERNMENT CASE AND THE JAVA PLATFORM PARADIGM

The Government, Netscape and Sun share a vision of a likely future for the computer software industry. In that future, internet servers will provide software to consumers, with the internet connection serving as the link between the application on the server and the user's home terminal.¹⁸⁶ The Java platform paradigm, identified in the Government's complaint in the current suit (and inter alia, in its memorandum in support of its contempt petition), is indeed one version of this future. It is not, however, the only possible version. The Government's challenge to Microsoft's marketing of its browser raises the question of how society will choose among alternative technologies. Although the Government has pointed to the Java platform paradigm as a possible future for the com-

sion of OS/2); Stanley Zarowin, *How to Pick the Perfect PC (For You)*, J. ACCT., Mar. 1995, at 65, 66 (discussing available hardware configurations).

182. See Gifford, *Microsoft Corporation*, *supra* note 33, at 651 (discussing backward compatibility of Windows 32-bit system); Howard Millman, *Windows 95 Headaches Start Bit by Bit: 32-Bit Applications Are Worth the Wait*, COMPUTERWORLD, Aug. 21, 1995, at 37, 37 (discussing virtues of Windows 95).

183. See Zarowin, *supra* note 181, at 66 (discussing hardware configurations available in 1995).

184. For a general discussion of Microsoft's technological developments, see Gifford, *Microsoft Corporation*, *supra* note 33, at 624-28.

185. See, e.g., Katz & Shapiro, *Systems Competition*, *supra* note 152, at 94 (discussing network effects and network externalities); Lopatka & Page, *Posner's Program*, *supra* note 164, at 1738 (explaining network externalities).

186. See Amy Cortese et al., *The Software Revolution*, BUS. WK., Dec. 4, 1995, at 78, 78 (describing software distribution through servers); see also *Sun Microsystems, Inc. v. Microsoft Corp.*, 999 F. Supp. 1301, 1302 (N.D. Cal. 1998) (explaining that Java technology gives software developers ability to create programming code capable of operating on many different, otherwise incompatible, system platforms).

puter software industry, the Government is advancing that paradigm cautiously. Its attack is cast as a challenge to a tying arrangement (that is also cast as monopolization and attempted monopolization)—the tie consists of the bundling by Microsoft of its browser with its operating system.¹⁸⁷ The challenge to the tie, however, has a much larger goal than the mere unbundling of the browser. It is designed to advance the likelihood that the Java platform paradigm replaces the Windows platform and thus ends Microsoft's dominance over operating systems.

In the Government's view, the question is whether Microsoft will be permitted to leverage its Windows monopoly into a monopoly over internet access.¹⁸⁸ Before addressing the Government's primary theory, let us look first at the Government's tying theory.

A. *Tying the Browser to the Operating System*

The Government has placed enormous weight upon its contention that Microsoft engaged in tying. Yet the tie employed by Microsoft is vastly different from conventional tying arrangements.¹⁸⁹ In the classic tie, the

187. See generally Complaint, *supra* note 1, ¶¶ 103-23, 134-37, 138-41.

188. See *id.* ¶¶ 13, 19 (alleging that Microsoft is using its power over operating systems to capture browser market); Memorandum, *supra* note 12, at 2 (arguing that Microsoft should be prevented from using "its monopoly power in operating systems to gain an unlawful advantage that could protect or extend its monopoly and inhibit competition"). See generally Thomas A. Piraino, *An Antitrust Remedy or Monopoly Leveraging by Electronic Networks*, 93 Nw. U. L. Rev. 1 (1998) (describing monopoly over internet access).

189. See Memorandum, *supra* note 12, at 12-15 (describing various aspects of Microsoft's tie-in arrangement between Internet Explorer and Windows 95). The controversy in the Microsoft antitrust case over the degree to which the Internet Explorer is a stand alone product or is integrated into the Windows operating system is diverting public attention away from the extent to which browsers and the operating system are coalescing together. See Halfhill, *Good-Bye*, *supra* note 70, at 60 (discussing trend of new products that will "integrate" web browsers and operating systems so that difference between them will be almost imperceptible to users). In the contempt proceedings before the district court, the parties were discussing the integration issue in terms of the extent to which the browser can be disabled without harm to the operating system. See *id.* (discussing numerous new products that will "blur the already increasingly irrelevant distinctions between native/cross-platform and local/remote applications"); see also Ronald S. Katz & Janet Arnold Hart, *Sideshow: U.S. v. Microsoft*, SC71 ALI-ABA 1, 6 (1998) ("A large portion of both the Government's and Microsoft's briefs were taken up with argument as to what constituted an 'integrated' product and what constituted a 'separate' product for purposes of the consent decree."). That question, however, involves only part of this broad convergence.

Both Microsoft and Netscape are responding to the perception that the public is increasingly demanding convergence. See Halfhill, *Good-Bye*, *supra* note 70, at 63 (characterizing trend towards web browser-operating system integration as "avalanche signif[y]ng a major trend"). The new Netscape Constellation inserts a shell around the Windows operating system so that the user stays within a single environment whether operating with material on the internet, intranets or in the user's own files on his or her personal computer. See *id.* at 66 (describing Netscape Constellation). In this case, the environment is the browser environment provided by Netscape. See *id.* Microsoft's Internet Explorer also provides a single environ-

seller reallocates some monopoly rents from the tying product to the tied product. Sellers often employ ties to facilitate price discrimination. Thus, in *IBM v. United States*,¹⁹⁰ involving early “computing machines” and punched cards, IBM leased its machines at a low lease rate and charged monopoly rates for the cards.¹⁹¹ By so doing, it charged high prices to high-intensity users (for whom the value of computing was high) and low prices to low-intensity users (for whom the value of computing was lower).¹⁹² The cards were, in effect, a metering device used to facilitate charging customers sums that varied with their use of the machines. Pricing by use is a common practice. Copying machines are commonly leased today under a lease arrangement that keys the rent to the usage of the machine. Although today’s auto rental companies are moving away from the practice, in the past they commonly based their charges on miles driven. It is not apparent that metering charges generally produce anticompetitive effects.¹⁹³

Metering or other forms of price discrimination may be objectionable, however, when carried out by a monopolist. Indeed, one of the grounds traditionally urged in support of the condemnation of tying arrangements has been their amenability to being used by a monopolist to discriminate in price. The metering charge involved in the *IBM* case could be visualized as price discrimination—a higher charge leveled against users with a higher demand than the charge leveled against users with a lower demand. Because IBM was the only producer of electronic

ment through its Active Desktop optional interface. *See id.* at 64-66 (describing Microsoft’s Internet Explorer and Active Desktop software). While the effect is visually similar to Netscape’s, Microsoft integrates the browser into the operating system rather than riding upon it, as in the Netscape version. *See id.* Sun Microsystems has integrated its browser into its operating system, apparently so thoroughly that the browser must be used to install the operating system. *See* Steve Hamm et al., *Does Everyone Do It?*, *Bus. Wk.*, Nov. 2, 1998, at 30, 30-31 (noting integration of operating system into single environment). When current developments in software are seen in the light of this new paradigm of a single user environment, these developments are better understood. Thus—contrary to the way the issue was presented in the contempt proceeding before the district court—the integration is not measured by the ability of the operating system to perform its standard functions when the browser is removed. Rather, the “integration” describes an improved browser/Windows-user interface, heightening the ability of Windows-users to access external material without leaving their normal operating environment.

190. 298 U.S. 131 (1936).

191. *See id.* at 136 (describing IBM’s leasing arrangements); *see also* ROBERT H. BORK, *THE ANTITRUST PARADOX* 377 (1978) (using facts of *IBM* to describe variable proportion price discrimination).

192. *See* BORK, *supra* note 191, at 377 (describing variable proportion price discrimination).

193. *See* Ward S. Bowman, Jr., *Tying Arrangements and the Leverage Problem*, 67 *YALE L.J.* 19, 23-24 (1957) (explaining why metering methods do not lead to new monopolies over tied products).

calculating machines at the time, that objection literally applied.¹⁹⁴ Observers, however, have pointed out that the more a monopolist employs price discrimination to charge each customer its reservation price (*i.e.*, the highest price that customer is willing to pay), the less the monopolist misallocates society's resources.¹⁹⁵ In fact, in the case of a perfectly discriminating monopolist, there is no restriction on output and no misallocation of resources.¹⁹⁶ If, as the Chicago School argues, efficiency is the sole goal of antitrust law, then the price discrimination carried out by a perfectly discriminating monopolist is consistent with that goal.¹⁹⁷ Outlawing IBM's attempt to replicate the behavior of a perfectly discriminating monopolist would engender incentives for it to behave as a single price monopolist, behavior that would produce the classic monopoly effect of misallocating resources.¹⁹⁸ In her concurring opinion in *Jefferson Parish Hospital District Number 2 v. Hyde*,¹⁹⁹ Justice O'Connor showed her awareness of the possibility that price discrimination might "decrease rather than increase the economic costs of a seller's market power."²⁰⁰ While she urged the abandonment of the *per se* rule against tying, she expressly avoided addressing the problem of price discrimination.²⁰¹ A majority of the Court, however, may still agree with Justice Stevens that an evil of tying

194. See *IBM*, 298 U.S. at 133. Remington Rand, Inc. also produced computing machines, but its machines were mechanically operated rather than electrically operated. See *id.*

195. See BORK, *supra* note 191, at 398 (asserting that "when the monopoly involved is lawful, the more the monopolist is permitted to discriminate, the better are the results in terms of resource allocation").

196. See *id.* (observing that discriminating monopolist would produce same output as competitive industries). It is possible, however, for a monopolist to discriminate and to reduce aggregate output beyond the output restriction associated with a single monopoly price, thereby exacerbating the misallocation of resources. See ROGER D. BLAIR & DAVID L. KASERMAN, *LAW AND ECONOMICS OF VERTICAL INTEGRATION AND CONTROL* 124 (1983) (asserting that price discrimination may contract as well as expand output).

197. See BORK, *supra* note 191, at 398 (noting that output of perfectly discriminating monopolist is same as output in competitive market). Although the Chicago School has been ascendant for several decades, some antitrust observers, while conceding that efficiency is a major goal of antitrust, deny that it is the sole or exclusive goal. See, e.g., Eleanor M. Fox, *Antitrust, Competitiveness, and the World Arena: Efficiencies and Failing Firms in Perspective*, 64 ANTITRUST L.J. 725, 730 (1996) (claiming that "efficiency, as opposed to competition, is not the *raison d'être* of antitrust law"); Robert H. Lande, *Wealth Transfers as the Original and Primary Concern of Antitrust: The Efficiency Interpretation Challenged*, 34 HASTINGS L.J. 65, 89-90 (1982) (propounding distributional standard for antitrust analysis).

198. See BORK, *supra* note 191, at 397-98 (describing possible adverse economic effects of prohibiting sellers from practicing price discrimination).

199. 466 U.S. 2 (1984).

200. *Id.* at 36 n.4 (O'Connor, J., concurring).

201. See *id.* (O'Connor, J., concurring) (arguing in favor of focusing on adverse and beneficial economic effects of tying instead of *per se* prohibition on tying, yet declining to address price discrimination facilitated by tying).

lies in its ability to facilitate price discrimination.²⁰² Under the classic view—reflected in Justice Stevens’s opinion in *Jefferson Parish*—a seller may exploit its monopoly by charging monopoly prices for the product over which it possesses a monopoly, but it may not shift monopoly rents from the tying product onto the tied product.²⁰³ To do so “extends” its monopoly unlawfully, and the resulting tie forecloses rival sellers from the market for the tied product.²⁰⁴

Finally, the per se rule against tying arrangements has also been justified as a means for ensuring that the seller will not use a monopoly position in the tying product to gain a new monopoly in the market for the tied product.²⁰⁵ This would involve the seller “leveraging” its power in the market for the tying product to generate a monopoly in the tied product market.²⁰⁶ There are few places, however, where such leveraging could occur. If there is a fixed proportional relationship between the two products, it is likely that a correct analysis would conclude that in fact only one product is actually involved.²⁰⁷ Indeed, normally in a case of fixed proportions no additional monopoly power can be acquired by the seller through tying.²⁰⁸ In many cases in which there is a variable proportion between the two products, the seller is trying to discriminate in price rather than to “monopolize” a second product.²⁰⁹

The current antitrust litigation does not fit especially well with the traditional reasons for condemning tying arrangements.²¹⁰ In this case,

202. *See id.* at 14-15 (asserting that tying impairs competition and facilitates price discrimination, thereby increasing monopoly profits).

203. *See id.* at 14 (stating that law draws distinctions between use of market power to charge monopoly prices for tying product, which is permitted, and use of market power over tying product to charge monopoly prices over tied product, which is prohibited).

204. *See id.* (asserting that tie could create barriers for new competitors’ entry into market for tied products).

205. This argument was common in the 1950s, when hostility towards tying arrangements was at its height. *See* CARL KAYSER & DONALD F. TURNER, *ANTITRUST POLICY: AN ECONOMIC AND LEGAL ANALYSIS* 157 (1959) (advocating subjecting tie-ins to per se rule because “[a] tie-in always operates to raise the barriers to entry in the market of the tied good to the level of those in the market for the tying good: the seller who would supply the one, can do so only if he can also supply the other”).

206. *See id.* (describing how sellers use power in market for tying good to gain competitive advantage).

207. *See* Donald F. Turner, *The Validity of Tying Arrangements Under the Antitrust Laws*, 72 HARV. L. REV. 50, 72 (1958) (“The sale of items used in fixed proportions might be deemed prima facie the sale of a single product . . .”).

208. *See* HERBERT HOVENKAMP, *FEDERAL ANTITRUST POLICY* 370-72 (1994) (explaining problems with leverage theory); Bowman, *supra* note 193, at 23 (observing that fixed proportions do not necessarily create monopoly).

209. For a further discussion of the use of tying as a means to discriminate in price, rather than to chill competition, see *supra* notes 189-208 and accompanying text.

210. *See* *United States v. Microsoft Corp.*, 980 F. Supp. 537 (D.D.C. 1997) (struggling with application of tying doctrine), *rev’d*, 147 F.3d 935 (D.C. Cir. 1998).

there does not appear to be any price discrimination.²¹¹ Therefore, concern about price discrimination cannot underlie the Government's enforcement action. Microsoft has been giving the browser away; it is not shifting any monopoly rents on to the browsers. Thus, contrary to *IBM*, there is no price discrimination. The Government, however, is casting its tying charge in the language of foreclosure and leveraging.²¹² These are indeed terms that traditional tying has repeatedly employed. Yet in the Microsoft context, these terms must be employed with care.

The Government sees Microsoft "foreclosing" Netscape and other potential browser sellers from the browser market.²¹³ Its current complaint repeatedly refers to this foreclosure.²¹⁴ Yet under the traditional foreclosure analysis, foreclosure is part of a scheme to shift rents to the tied product, a scheme that is not present here. The Government complains both that Microsoft has given away its browser and that it is tied to the operating system.²¹⁵ These practices have foreclosed Netscape from selling its product, thus depriving it of substantial revenues. But Netscape was also giving its product away.²¹⁶ These circumstances provide an unusual context in which to apply traditional tying doctrine. That doctrine fits uneasily into a framework in which both major players have, at one time or another, engaged in the practice of giving away their products. Giving the product away freely is not tying. Is bundling different from giving the product away freely? Perhaps, but the difference needs to be explained.

It is a version of the leveraging objection, however, that really underlies the Government's concern about tying in the current antitrust case. Indeed, it was that same version of leveraging that underlay the Government's contempt proceeding.²¹⁷ But it did not fit the circumstances of the contempt proceeding very well. First, in the contempt proceeding, the Government was only challenging Microsoft's bundling of its browser with Windows 95, a bundling that has been superseded by Microsoft's full inte-

211. Because Microsoft is not charging separately for the browser, there is no price discrimination. For a further discussion of the practice of giving browsers away, see *supra* notes 112-25 and accompanying text.

212. See Complaint, *supra* note 1, ¶¶ 12, 18, 22, 75, 97, 103(h), 111, 123, 127, 137. A leading critic of Chicago-School leveraging analysis is Professor Louis Kaplow. See Louis Kaplow, *Extension of Monopoly Power Through Leverage*, 85 COLUM. L. REV. 515, 515 (1985) (discussing practice of tying in conjunction with leveraging).

213. See, e.g., Complaint, *supra* note 1, ¶ 103(R) (alleging effort by Microsoft "to foreclose opportunities for non-Microsoft browsers to establish themselves").

214. See *id.* ¶¶ 22, 111, 123, 127, 137 (alleging foreclosure).

215. See *id.* ¶¶ 16, 22, 23, 103, 108, 109-12, 115-21, 127, 135-41 (alleging tie and failure to impose separate charge).

216. For a discussion of the practice of competitors giving away browsers, see *supra* notes 112-25 and accompanying text.

217. See Memorandum, *supra* note 12, at 29; see also Daniel L. Rubinfeld, *Antitrust Enforcement at DOJ: An Economist's Perspective*, Address Before The Association of the Bar of the City of New York (Nov. 17, 1997), available in 1997 WL 792881.

gration of the browser and the operating system in Windows 98. It thus appears implausible to argue that the temporary bundling of the browser with Windows 95 between the district court's order of December 1997 and May 1998 (when Windows 98 was released) could generate a monopoly for Microsoft over internet browsers. Second, there are numerous problematic aspects of its tying case under the traditional doctrinal analysis applicable to tying arrangements.²¹⁸

These problematic aspects of the contempt proceeding may have been rendered effectively moot by the new antitrust case. In the new suit, the Government is attacking ties of the browser to both Windows 95 and Windows 98 and contending that these ties are designed to help Microsoft acquire a monopoly in the browser market.²¹⁹ Even here, however, the claim that Microsoft is attempting to acquire a monopoly over the browser market is an odd claim, given that browsers are being given away. If Microsoft is tying the browser to the operating system by selling the two as one product, how is it increasing its profits by doing so?

The Government, however, does not object to the tie for completely traditional reasons.²²⁰ The Government's objection is about leveraging, but it is a peculiar form of leveraging that makes use of the relatively new concepts of network effects and path dependency theory.²²¹ Even without these new theories, a leveraging case might exist when the users of the tying product are the principal customers for the tied product and the products are not purchased in a fixed proportion.²²² Such a circumstance would be an unusual one, but that relationship may exist between users of operating systems and customers for browsers. The Government seems to be identifying the bundle of the browser with the operating system as an arrangement that fits this profile and then has added to its formulation a time dimension along with the network and path dependency analyses.

The Government, under this approach, objects to the tie, viewing it as a device by which Microsoft utilizes the network effects phenomenon to

218. For a discussion of traditional tying cases, see *infra* notes 225, 226, 232 and accompanying text.

219. See Complaint, *supra* note 1, ¶¶ 103-12 (alleging Windows 95 tie); *id.* ¶¶ 113-23 (alleging Windows 98 tie).

220. For a further discussion of the Government's basis for objecting to the tie between the Explorer and Microsoft servers, see *supra* note 71 and accompanying text.

221. There is ground for believing that the Government's tying claim is, at base, one of temporal tying. For a discussion of temporal tying, see *infra* notes 223-24 and accompanying text. See Memorandum, *supra* note 12, at 13-39 (detailing Government's leveraging charges against Microsoft).

222. See RICHARD A. POSNER, ANTITRUST LAW: AN ECONOMIC PERSPECTIVE 173 (1976) ("It is only when the users of the tying product are also the principal customers for the tied product—by no means the typical tie-in case—that tying can rightly be described as a method of obtaining a second monopoly."). Although Posner then questions why a firm would want to obtain a monopoly on a complementary good, he does not deal with the circumstance in which a second monopoly replaces an eroding monopoly. See *id.*

establish its dominance over internet technology.²²³ Here, the tie produces its effects temporally, rather than in the static market employed in the traditional analysis.²²⁴ The Government's case is therefore a combination of the doctrinal with newly emerging economic theory.

But even in a temporal analysis, we have to consider the "power" of the seller of the tying product to coerce consumers to take an unwanted product in substitution for a desired product.²²⁵ Here, there are some more troublesome aspects of the Government's argument. The imposition of a tie requires that a buyer be "forced" or coerced to take a product that it does not want.²²⁶ The tie appears to be Microsoft's insistence that computer manufacturers who choose to install its operating system also install its browser.²²⁷ There are grounds for believing that Microsoft does not object to a manufacturer adding the Netscape or other browser so long as Microsoft's browser is also included.²²⁸ If this is the case, then it is difficult to see how there could have been even a technical tie.

In addressing the issue of tying, the courts have failed to define adequately the full scope of behavior that can constitute a tying arrangement.²²⁹ Congress, however, has defined tying arrangements in section 3 of the Clayton Act²³⁰ by focusing on foreclosure.²³¹ Surely this definition should be persuasive in determining what constitutes a tie for purposes of

223. See Complaint, *supra* note 1, ¶ 127. For a discussion of network effects as applied to Microsoft, see Lemley & McGowan, *Legal Implications*, *supra* note 164, at 500-07 (discussing effects of forced integration under network effects phenomenon).

224. See Lemley & McGowan, *Competitive Propriety*, *supra* note 12, at 733-50 (discussing "temporal tying").

225. See *IBM v. United States*, 298 U.S. 131, 139-40 (1936) (observing that other cards of equal or better quality may be offered at same or lower prices).

226. See *Jefferson Parish Hosp. Dist. No. 2 v. Hyde*, 466 U.S. 2, 32 (1984). In *Jefferson Parish*, the Court stated:

[T]he essential characteristic of an invalid tying arrangement lies in the seller's exploitation of its control over the tying product to force the buyer into the purchase of a tied product that the buyer either did not want at all, or might have preferred to purchase elsewhere on different terms. When such "forcing" is present, competition on the merits in the market for the tied item is restrained and the Sherman Act is violated.

Id. at 12.

227. See Complaint, *supra* note 1, ¶¶ 103-23 (alleging unlawful tying of browser to Windows 95 and 98 operating systems); *United States v. Microsoft Corp.*, 147 F.3d 935, 945-53 (D.C. Cir. 1998) (considering tying issue in contempt proceeding).

228. See generally *Senate Hearing*, *supra* note 10 (noting testimony of Bill Gates).

229. See Daniel J. Gifford, *The Damaging Impact of the Eastman Kodak Precedent Upon Product Competition: Antitrust Law in Need of Correction*, 72 WASH. U. L.Q. 1507, 1510-11 & nn. 7, 8 (1994) [hereinafter Gifford, *Damaging Impact*] (discussing Supreme Court's lack of clarity in *Eastman Kodak Co. v. Image Technical Services, Inc.*, 504 U.S. 451 (1992)).

230. 38 Stat. 730 (1914) (codified as amended at 15 U.S.C. § 14 (1994)).

231. Section 3 prohibits tying that may "substantially lessen competition or tend to create a monopoly in any line of commerce." *Id.*

the Sherman Act as well. The Sherman Act cases, as well as the Clayton Act cases, reflect a similar concern with foreclosure.²³² In the new anti-trust case, the Government has made numerous assertions that Microsoft's tying practices were foreclosing the browser market to Netscape and others.²³³ But so long as Microsoft allows computer manufacturers to install non-Microsoft browsers in addition to the Explorer, it is difficult to identify foreclosure. The manufacturer is free to install whatever additional browser it wishes, and when the manufacturer installs an additional browser consumers will have the ultimate choice as to which browser to use.²³⁴ Thus, the foreclosure analysis associated with traditional tying doctrine does not quite fit in this case.

In the traditional tie, the purchaser is prevented from acquiring the tied product from an alternative source because the purchaser has already paid the tying seller for the tied product. To acquire the tied product from an independent source would require the purchaser to pay for its supplies twice. That kind of effect inhered in the lump-sum licenses that Microsoft had entered into with computer manufacturers prior to the consent decree.²³⁵ Having paid a lump-sum for a license to install as many Windows operating systems as it needs, the computer manufacturer is positively discouraged from seeking a license from an alternative supplier of operating systems. To do so would require the manufacturer to pay twice.²³⁶ In the browser market, however, this is not the case. The computer manufacturer who is furnished a free copy of the Explorer may still acquire an alternative browser (such as the Navigator) at no greater cost than it would incur had it purchased the Navigator in the absence of the bundle. Indeed, now that Netscape is giving away its browser, computer manufacturers can add its browser to their products without cost. There is no obvious foreclosure here.

232. See, e.g., *Eastman Kodak Co.*, 504 U.S. at 478 (viewing "market foreclosure" as facially anticompetitive in context of tying claim); *Northern Pac. Ry. v. United States*, 356 U.S. 1, 6 (1958) (observing that tying arrangements "deny competitors free access to the market for tied product" and force buyers to forego their free choice between competing products).

233. See Complaint, *supra* note 1, ¶¶ 23, 103(h), 111, 123, 127, 137.

234. Of course, in the case described, Microsoft has arguably affected the choice of the computer manufacturer. In the absence of the bundling, the manufacturer might have decided to install only the Netscape Navigator. Even so, it is hard to identify foreclosure so long as the manufacturer remains free also to install the Navigator. It may be said that the manufacturer will not install both the Navigator and the Explorer and once it is required to take the Explorer, it will no longer want the Navigator. Perhaps. But why not? If a computer manufacturer believes that their customers prefer the Navigator, an economically rational computer manufacturer would install the Navigator. Cf. *id.* (finding that superior browser will ultimately win).

235. See Gifford, *Microsoft Corporation*, *supra* note 33, at 632-33 (describing objectionable features of lump-sum licenses).

236. See *id.* (discussing double payment consequences inherent in lump-sum license); see also Baseman et al., *supra* note 15, at 281-82 (discussing increased costs associated with lump-sum licenses).

Perhaps, however, there is an indirect foreclosure as a result of additional costs that computer manufacturers would incur in installing a second browser—computer manufacturers might decide that they want only one browser on the machines they produce because installing two browsers would increase their costs for technical support. Perhaps. It is the number of computer purchasers using browsers, however, that seems to constitute the relevant datum for purposes of technical support. Most users probably stick with a single browser. It does not appear likely, therefore, that the installation of additional browsers would generate a surfeit of additional calls for technical help. If the costs of installing an additional browser are in fact minimal, this indirect foreclosure becomes less easy to identify.

Not only is it difficult to see foreclosure in the bundling, it is similarly difficult to see the difference between Microsoft's bundling the browser with its operating system and its giving away the browser to anyone who wants it. In both cases, the browser is provided free-of-charge. Indeed, during the period that the district court had enjoined Microsoft from bundling the browser with the operating system, these alternatives coalesced. Microsoft complied with the injunction by offering computer manufacturers the option of taking the Windows 95 operating system with an operative browser or of disabling it.²³⁷ That choice was effectively whether to accept a free browser or to reject it. So long as Netscape was charging a fee for the browser, Microsoft's free offer was probably especially difficult to refuse, at least if—and to the extent that—the Microsoft browser is qualitatively comparable to the Netscape browser.

If Microsoft had not only required computer manufacturers to include its browsers on their machines, but also required them to exclude the Netscape Navigator, then the preceding analysis would not apply. Under these circumstances, a plausible case of forcing would be stated and the Government would have a valid basis for demanding an end to this exclusionary practice. Yet in the contempt proceeding, the injunction was cast in terms of ordering Microsoft to cease bundling, rather than in terms of ordering Microsoft to refrain from prohibiting the installation of rival browsers on newly manufactured computers.²³⁸ This was an odd phrasing if the exclusionary behavior consisted of an insistence that rival browsers be barred from new Windows-licensed machines. If Microsoft had insisted only on bundling, but did not object to manufacturers adding other browsers to their machines, then there was no obvious exclusion and the injunction itself requires justification—a justification not found in (or with) the court's order.²³⁹

237. See *United States v. Microsoft Corp.*, 147 F.3d 935, 940-41 (D.C. Cir. 1998) (discussing injunction).

238. See *United States v. Microsoft Corp.*, 980 F. Supp. 537, 545 (D.D.C. 1997), *rev'd*, 147 F.3d 935 (D.C. Cir. 1998).

239. See Daniel J. Gifford, *The Jurisprudence of Antitrust*, 48 SMU L. REV. 1677, 1711-12 (1995) [hereinafter Gifford, *Antitrust*] (arguing for concerted efforts by

In the antitrust suit, the Government takes the view that Microsoft may bundle the browser with its operating system, but subject to two major conditions. First, the Government wants computer manufacturers to be able to delete the Microsoft browser and to be compensated for any costs incurred in the deletion.²⁴⁰ Second, for three years a Microsoft integrated operating-system/browser bundle must include the rival Netscape browser and computer manufacturers must be given the option of deleting the Microsoft browser if they so desire.²⁴¹ The Government's primary focus thus appears to be on enabling computer manufacturers to delete the Microsoft browser. But the Government's acceptance of bundling, provided that the Netscape browser is included in the bundle, transforms a related aspect of its position from hostility to bundling to an acceptance of bundling, so long as the bundling does not foreclose. If in fact the present licensing system does not preclude computer manufacturers from adding additional browsers, the difference between that aspect of the relief sought and the present practice is narrow. Under present practice, the computer manufacturer lacks the option of deleting the Microsoft browser and it is at least an open question as to whether a computer manufacturer is discouraged from adding a second browser by the prospect of incurring additional support or other costs.

The bundling analysis, however, can be rendered moot by another aspect of traditional tying doctrine—the requirement that there be two products, a tying product and a tied product. The question of whether one or two products is involved has been a bane of the tying aspect of antitrust for decades. Over forty years ago in one of the foundation cases of tying law, *Times-Picayune Publishing Co. v. United States*,²⁴² the Government floundered on this requirement.²⁴³ In that case, the Government contended that the defendant newspaper publisher had been tying advertising in its morning newspaper to advertising in its evening paper.²⁴⁴ The Government lost, however, as the Supreme Court ruled that advertising in the two papers constituted a single product.²⁴⁵

enforcement authorities and courts to advance coherence as antitrust goal). This particular phrasing of the injunction carries the potential for weakening public confidence in enforcement institutions. If the Government and the court believe that the bundling of a free item forecloses, both institutions should explain their reasoning.

240. See Complaint, *supra* note 1, at Part VIII, ¶ 1(f) (requesting decree provision enabling computer manufacturers to delete Explorer icon and to disable Explorer).

241. See *id.* ¶ 1(e) (requesting decree provision requiring bundling of Netscape browser whenever Explorer is bundled).

242. 345 U.S. 594 (1953).

243. See *id.* at 614 (“[T]wo newspapers under single ownership at the same place, time, and terms sell indistinguishable products to advertisers; [and therefore] no dominant ‘tying’ product exists . . .”).

244. See *id.* at 600.

245. See *id.* at 613.

A peculiarly difficult form of this issue involves the integration of previously separate items into a new single product. During the 1970s, the lawfulness of IBM's integration of previously separate peripheral devices into its mainframe computers reached the courts, albeit sometimes under the rubric of monopolization. In a leading case from that period, *California Computer Products, Inc. v. IBM Corp.*,²⁴⁶ the United States Court of Appeals for the Ninth Circuit ruled that even a monopolist has "the right to redesign its products to make them more attractive to buyers—whether by reason of lower manufacturing cost and price or improved performance."²⁴⁷ Subsequently, in *Memorex Corp. v. IBM Corp.*,²⁴⁸ the Ninth Circuit relied upon its earlier decision in *California Computer Products* when it affirmed a district court determination that a "head/disk assembly" was a component part in a disk drive and thus did not constitute a separate product for tying purposes.²⁴⁹ In effect, the Ninth Circuit ruled that the rationale which it had adopted to evaluate product integration for monopolization purposes applied as well to issues of technological tying.²⁵⁰ That court's subsequent decision in *Foremost Pro Color, Inc. v. Eastman Kodak Co.*²⁵¹ employed analogous reasoning—after ruling that Kodak had not been engaged in unlawful technological tying, the court ruled that the substance of the unsuccessful tying claim could not be used to support a monopolization claim.²⁵²

The tying issue was addressed last June by the United States Court of Appeals for the District of Columbia in the contempt proceeding.²⁵³ Following an analysis in the Areeda antitrust treatise,²⁵⁴ the court ruled that

246. 613 F.2d 727 (9th Cir. 1979).

247. *Id.* at 744.

248. 636 F.2d 1188 (9th Cir. 1980).

249. *See id.* ("We are unable to distinguish Memorex's case from the *California Computer Products* case, and we conclude that, on the authority of that case the judgment should be affirmed.").

250. *See* ILC Peripherals Leasing Corp. v. IBM Corp., 448 F. Supp. 228, 233 (N.D. Cal. 1978) (explaining requirements for illegal tying arrangement). Generally, the courts have rejected claims of technological tying and/or monopolization where a product integration appears to offer improved convenience or value. *See, e.g., California Computer Products*, 613 F.2d at 744 (discussing benefits of product integration); *Response of Carolina, Inc. v. Leasco Response, Inc.*, 537 F.2d 1307, 1330 (5th Cir. 1976) (noting that beneficial technological result may warrant tying); *Innovation Data Processing, Inc. v. IBM, Corp.*, 585 F. Supp. 1470, 1476 (D.N.J. 1984) (finding arrangement in question to be "lawful package of technologically integrated components"). Courts have also rejected technological tying and associated monopolization claims in contexts in which a newly developed technology required several new products to be used in conjunction with each other. *See Foremost Pro Color, Inc. v. Eastman Kodak Co.*, 703 F.2d 534, 542-43 (9th Cir. 1983) ("We do not believe that, standing alone, such technological interrelationship among complementary products is sufficient to establish the coercion essential to a per se unlawful tying arrangement.").

251. 703 F.2d 534 (9th Cir. 1983).

252. *See id.* at 543-45.

253. *See* United States v. Microsoft Corp., 147 F.3d 935, 935 (D.C. Cir. 1998).

254. 10 PHILLIP E. AREEDA ET AL., ANTITRUST LAW ¶ 1746b (1996).

an "integrated product" is "a product that combines functionalities (which may also be marketed separately and operated together) in a way that offers advantages unavailable if the functionalities are bought separately and combined by the purchaser."²⁵⁵ As the court saw it, the Explorer and the Windows operating system shared such substantial amounts of overlapping code that they met one part of this test.²⁵⁶ It was Microsoft, the manufacturer, which wrote the code that was used both in the browser and in the operating system.²⁵⁷ Neither the end-user nor the computer manufacturer that installed these systems on the computers could combine these separate functionalities.²⁵⁸ They were combined at the manufacturing stage in the code design.²⁵⁹ As to the issue of whether the combination offered advantages to purchasers, the court ruled that in interpreting the consent decree "the question is not whether the integration is a *net* plus but merely whether there is a plausible claim that it brings some advantage."²⁶⁰ Although the court left open the question as to whether the latter standard is appropriate for antitrust law generally, the court relied heavily on antitrust authorities and indicated throughout its opinion its belief that courts lack institutional competence to review design decisions.²⁶¹ Moreover, the court took pains to point out that its analysis was consistent with antitrust tying law.²⁶² The logic of the court's approach suggests that for antitrust law generally, courts should adopt a deferential rationality standard, accepting the manufacturer's design decisions so long as those decisions plausibly provide advantages to customers.²⁶³ The basic approach set forth in this opinion will probably guide the disposition of the tying issues in the new antitrust case. The district court, despite its own "misgivings," has acknowledged that the court of appeals decision of last June provides support to the Microsoft defense.²⁶⁴

The analytical framework set forth by the court of appeals is an attractive one because it avoids the problems of courts second-guessing manufacturers on product design. Yet that framework does not provide a basis for addressing the issue of most concern to the Government—that of the

255. *Microsoft Corp.*, 147 F.3d at 948.

256. *See id.* at 949.

257. *See id.*

258. *See id.*

259. *See id.* (discussing code combination offered by manufacturer).

260. *Id.* at 950.

261. *See id.* at 948-51.

262. *See id.* at 950 (discussing precedent and concluding that its own analysis is "consistent with tying law").

263. *See id.* at 948 ("Antitrust scholars have long recognized the undesirability of having courts oversee product design, and any dampening of technological innovation would be at cross-purposes with antitrust law.").

264. *See United States v. Microsoft Corp.*, 1998-2 Trade Cas. (CCH) ¶ 72,261, at 82,672-82,677 (D.D.C. 1998) (noting that United States Court of Appeals for District of Columbia Circuit adopted "technological tying" standard proposed by Microsoft).

relation between the Java programming language and the Microsoft monopoly over operating systems. Suppose that we can avoid the difficult doctrinal problems presented by the tying, monopolization and attempted monopolization case law, so that we can face the policy issue directly. When a business firm holding an upstream monopoly wants to integrate its operations downstream, should it be required to follow a two-track scenario by offering the public—along with its integrated product—the option of an unintegrated upstream product as well? That is essentially what the Government is demanding and the district court is inclined to grant.²⁶⁵ The benefits of such an approach would, of course, be that the public would have the option of accepting or rejecting the integration. The downside is that once that option were imposed, the producer would be under continuing governmental and judicial pressure to ensure that the upstream product remained compatible with rival downstream products.

In context, this would mean that Microsoft would be permitted to integrate the Explorer into its operating system only if it also makes an unintegrated version of the operating system available. An operating system in which the Explorer can be deactivated and replaced with an independent browser appears to be acceptable to the Government. This scenario, however, seems to assume that the operating system and the independent browsers will be fully compatible. Would this scenario require those aspects of the operating system that interact with browsers to be frozen at their current stage of development? Would Microsoft, under such a scenario, be required to coordinate its operating system development with Netscape to ensure that changes in its operating system remained compatible with the Netscape browser and that changes in the Netscape browser interfaced properly with the operating system? Neither the Government nor the district court have addressed these aspects of this two-track scenario.

These are intriguing and difficult issues. They force us to face fundamental values. We want the courts to guarantee market freedom, yet we do not want business decisions—including those involving product design—to be made by the courts. We face here, on the one hand, the limits of judicial competence and, on the other hand, a determination of what market freedom means. It is possible, however, that the present litigation does not require these issues to be resolved now. If, as appears likely, the underlying Government concern is the possibility that Microsoft will use its Windows monopoly to prevent the emergence of the Java paradigm, then that concern may be put to rest without impinging upon the integration of the Explorer into the Windows operating system. The remainder of this Article argues that the temporal leveraging analysis underlying

265. *See id.* at 82,674 (“The market can make that determination only if two bundled products are also offered in their unbundled form, as they were in the IBM cases.”).

much of the Government's two antitrust proceedings is flawed. It is flawed because it relies upon a network-effects scenario that the Government may not have fully analyzed.

B. *Innovation and Incompatible Systems*

Although the Government's contempt charges against Microsoft can be evaluated under traditional tying doctrine, it would unjustly depreciate the Government's case to take that approach. Classic tying doctrine is heavily grounded upon an archaic view of economics. In neither the contempt proceeding nor the subsequent antitrust case can the Government be accused of relying on outdated economic analysis or an excessively doctrinal approach towards antitrust law. Rather, the Government is relying upon new theories of economics dealing with network externalities and a new vision of the computer industry predicated upon the success of the Java platform paradigm. It is this vision and these new analytical tools that are critical to the Government's strategy.

In seeking to prevent Microsoft from tying the browser to its operating system, the Government seeks to expand the opportunities for the Java paradigm to evolve. Currently, several of the market's major actors are advancing this paradigm, relying on Java as a competitive weapon against Microsoft. Thus Sun Microsystems has developed, and is promoting Java, not for revenues that Java licenses can generate for it directly, but as a means for projecting its image as a leader in the computer industry, and perhaps as a means for staving off the replacement of its low-end work stations running on Sun's Solaris, its UNIX-based operating system, with cheaper personal computers running on Intel microprocessors and Windows NT.²⁶⁶ Also, Netscape has committed itself to Java, hoping that through Java, its browser will become the user interface of the future, replacing the Windows interface now controlled by Microsoft.²⁶⁷ Indeed, Netscape has developed technology to cast a shell over the Windows interface so that users will employ the Netscape browser as the control panel not only for interacting with the internet, but for interacting with programs on their hard disks as well.²⁶⁸ The Government, however, remains concerned that despite its backing by Sun Microsystems, Netscape, IBM

266. See Steve Hamm & Robert D. Hof, *Operation Sunblock: Microsoft Goes to War*, BUS. WK., Oct. 27, 1997, at 110, 110, 112 (discussing possibility of Microsoft assault on Sun workstation market); David Kirkpatrick, *Meanwhile, Back at Headquarters . . .*, FORTUNE, Oct. 13, 1997, at 82, 82-84 (describing possible threat to Sun Unix-based servers from Windows NT).

267. See Halfhill, *TODAY the WEB*, *supra* note 49, at 68 (asserting that Java could surpass Windows as software platform by end of century).

268. See David Bank, *Microsoft Moves to Control the PC Screen*, WALL ST. J., Dec. 5, 1996, at B2 (discussing Netscape's shell technology, known as Constellation, for use on top of Windows). For a discussion of Netscape's apparent failure to follow up on its Constellation technology, see Mark Schlack, *Byte*, July 1998, at 12 ("Where is the web top that Netscape teased us with when it launched the Constellation beta a year ago?").

and others, the Java paradigm may well be the superior technology that is defeated by the network effects of Microsoft products.²⁶⁹ Because of this concern, the Government is aggressively attempting to ensure that Microsoft does not leverage its advantage in operating systems into dominance of internet technology.

The rivalry and conflict between Netscape, Sun, IBM and others with Microsoft, as well as the antitrust proceeding brought by the Government, are connected in one way or another with issues of software compatibility and/or interoperability. Many, but not all, of these conflicts are connected to the vision of the Java platform paradigm. Despite their superficial similarities, these actual conflicts, as well as other potential conflicts, raise different competitive issues and, therefore, need to be separately identified and evaluated. This is the subject of the next section.

V. HOW INNOVATION AND INCOMPATIBILITY RELATE TO THE JAVA PLATFORM PARADIGM

The vision of the Java platform paradigm is attractive because it allows all software applications and operating systems to interact.²⁷⁰ Indeed, this vision of a future universal compatibility—a compatibility dependent upon Java's interoperability feature—appears to be a major advance over the present internet system. The present system lacks interoperability between platforms, but is nonetheless a world in which universal accessibility to internet sites can be actively pursued as an ideal and approached in practice. Microsoft and Netscape actively compete in the browser and server markets, but consumers expect the browsers of both companies to provide them with access to virtually all internet sites. As the discussion below indicates, although universal access is not invariably achieved, it remains nonetheless an ideal in the minds of consumers. As a result, both companies are constrained by this ideal.

Both the ideal of present universal accessibility and the vision of a future in which all software can interact appear to be threatened when either of these major companies (1) markets browsers that are incompatible with widely employed server software, (2) markets server software that is incompatible with the browser provided by the other company or (3) markets either browsers or servers that employ nonstandard versions of Java.

269. See Complaint, *supra* note 1, ¶¶ 68, 127 (relating allegations of tying to Java and network effects).

270. See Lemley & McGowan, *Competitive Propriety*, *supra* note 12, at 733-34 (finding platform innovative because it may represent development of operating system allowing users to access other software applications in addition to Windows).

A. *Two Kinds of Incompatibility*

Underlying concerns about software incompatibility on the internet are two significantly different kinds of incompatibility. Both kinds of incompatibility involve the products of Netscape and Microsoft in the server/browser market. These two kinds of incompatibility have very different ramifications for the software industry and the consumers of its products. The first broad kind of software incompatibility grows out of the intense rivalry between Netscape and Microsoft and tends to be temporary in nature. Being temporary in nature, it carries no threat to the long-term universality of internet standards. The second kind of software incompatibility is of a deeper, structural kind that is more likely to last. It carries ramifications for the direction in which internet standards develop. Structural incompatibility directly implicates the future of the Java platform paradigm. The societal ramifications of the rivalry engendering these different types of incompatibilities are addressed below.

1. *Innovation and the Process of Standardization in a Context of No-Holds Barred Competition*

Microsoft and Netscape vigorously compete for the servers and browsers markets.²⁷¹ The result of this competition is a constant stream of mutually incompatible innovations in server/browser technology.²⁷²

A lucrative sector of the server/browser market is the intranet, the private networks established by business firms and groups of corporate affiliates for internal communication.²⁷³ The prevailing understanding of the industry is that a seller's server/browser packages become increasingly attractive to intranet buyers as its share of the installed base of browsers increases.²⁷⁴ This belief helps explain the willingness of Microsoft and

271. See Angela Hickman, *E-Commerce: The Missing Link*, PC MAG., Apr. 7, 1998, at 29 (noting that Microsoft and Netscape are main contenders in web server market); Steve Lohr, "Browser War" Limits Access to Web Sites, N.Y. TIMES, Dec. 8, 1997, at D1 (acknowledging market share and innovation war between Microsoft and Netscape).

272. See Lohr, *supra* note 271, at D1 (discussing inability to access entire web because of incompatibility of Netscape and Microsoft technology).

273. See Ian C. Ballon, *Intellectual Property Protection and Related Third Party Liability*, 482 PLI/PAT 559, 645 (1997) (defining intranet as "an employer's computer network"). According to Netscape Chief Executive James L. Barksdale, the browser war is the means to a more lucrative end—the sale of server software for corporate intranets. See Robert D. Hof et al., *Cyberspace Showdown*, BUS. WK., Oct. 7, 1996, at 34, 34 (reporting that Netscape is competing actively for corporate intranet software business); Laurence Zuckerman, *Browser Moves by Microsoft Make Even Netscape Blink*, N.Y. TIMES, Oct. 9, 1996, at D1 (describing battle over server software used by corporate clients for internal Web-like networks called intranets).

274. See Green, *supra* note 113, at 69 (stating that Netscape "pioneered the internet business model" of giving away software to build up market share).

Netscape to give away their browsers.²⁷⁵ Their desire to foster sales to the intranet market also helps explain their intense rivalry in innovative activity.²⁷⁶ Competition for intranet sales drives each company to devise a continuous series of improvements for its products, each trying to outperform the other.

The result of this continuous stream of improvements and added capabilities from each company is that the latest developments by Microsoft will not be compatible with the latest versions of Netscape and vice versa. The incompatibility results because each company tries to get a lead on the other. The principal technological divergences resulting from the rivalry between Netscape and Microsoft take the form of differing standards governing Java, differing enhancements to HTML and similar matters whose evolution is ultimately guided by a standards body. This incompatibility at the cutting edges of innovation does not pose a problem in much of the intranet market where the users—particular companies or corporate networks—purchase browser/server software for their entire operation.

Carried over the internet, however, this intense rivalry in innovation means that the most recent innovations in server/browser software are not likely to be compatible with major parts of the installed base of existing browsers.²⁷⁷ Of course, it is possible for users, at nominal or no cost, to obtain the latest versions of both of the principal browsers.²⁷⁸ By doing so, a user could receive communications from any server, even if that server employed the technology incorporated in the latest releases of Netscape or Microsoft.²⁷⁹ Most browser users, however, do not take the trouble to maintain two browsers in their most recently revised form.²⁸⁰ Because server site operators recognize the incompatibility of much of the installed bases of browsers with the most recent releases of either Netscape or Microsoft, they do not use those most recent releases on internet server

275. *See id.* (describing computer companies' incentive for providing free browsers to computer manufacturers as means of increasing number of its own browsers on market).

276. *See* Hof et al., *supra* note 273, at 34 (discussing Netscape's plan to use new technology to rival Windows).

277. *See* Lohr, *supra* note 271, at D1 (stating that internet web developers must often design software to fit either Netscape's Navigator or Microsoft's Explorer to employ latest server/browser innovations). Although there are a few server sites that remain completely inaccessible to one kind of browser, a more common problem is for a portion of a server site to malfunction with a particular browser. *See id.* (explaining that recent browser innovations typically produce denial of accessibility to portions of server sites as opposed to total exclusion). Most sites, however, are fully accessible to all browsers. *See id.*

278. *See* Green, *supra* note 113, at 69 (noting that Microsoft and Netscape have made browsers available for downloading cost free).

279. *See* Lohr, *supra* note 271, at D1 (observing that users could install both browsers as vehicles for accessing entire web).

280. *See id.* at D13 (describing disadvantages of installing duplicative browsers). Loading two browsers, inter alia, consumes space on a PC's hard drive. *See id.*

sites. As a result, the implementation of innovative technology is often postponed for a year or more, until Netscape and Microsoft work out their technological differences.²⁸¹ Nonetheless, both companies appear to recognize the need to maintain universal access to the internet. Therefore, these incompatibility problems tend to be temporary ones, resulting from competitive pressure to innovate, pushing one or both companies technologically ahead of existing protocols.²⁸² Ultimately, the protocols are modified and both companies do what is necessary to bring their modifications into compliance, thus preserving universal access.²⁸³

The market generates both parts of this scenario—it fosters the rivalry in innovation that generates the temporary incompatibilities and it pressures the two companies to resolve these incompatibilities in cooperation with the relevant standards organizations. As long as each company accounts for a major part of the installed base of browsers, neither company can ignore the problem of compatibility. Browsers that were compatible with only a portion of server sites would be problematic in the eyes of consumers, as would server software that was incompatible with a major portion of the browsers currently in use.

The model describing innovation in server/browser software is a dynamic one in which each company seeks to achieve an edge over the other by providing either a more convenient or technologically superior product. As one company obtains an edge, it exploits that edge in sales of server/browser software in the intranet market where the homogeneity of networks can be assured. Most internet server sites do not employ the most recent innovations in server/browser technology despite their availability to those sites and their programmers. The result is a steady stream of innovations for the intranet market that, although not immediately usable for the internet, ultimately become fully usable in that market as well.

An alternative and more orderly model exists in which changes in official standards precede the implementation of new technology. This model resembles the one propounded by Sun for Java—a core model of full interoperability through a Java-based and Sun-supervised set of standards. Even this model recognizes a penumbral area in which Java practitioners do not always achieve the full interoperability that they desire. That penumbral category is probably a practical necessity, as companies developing Java capabilities will almost invariably do so ahead of the next

281. *See id.* (“Web developers delay adding cutting-edge technology to their sites until Microsoft and Netscape agree to make their browser technology compatible . . .”).

282. *See id.* (discussing how competitive pressures may prompt companies to correct incompatibility problems). At the recent Senate Hearings, both Microsoft CEO Bill Gates and Netscape CEO Jim Barksdale asserted their intention to reconcile their newly developed technology with internet standards set by bodies such as W3C. *See generally Senate Hearing, supra* note 10 (discussing efforts by computer companies to create compatible browsers).

283. *See Lohr, supra* note 271, at D13 (discussing fact that cooperation by computer companies ultimately will realize universal internet access).

set of standards. The flexibility of this model at the edges facilitates development. Development is further fostered by the ability of Sun's Java licensees to acquire intellectual property rights to their improvements, even if those rights would be subordinate to Sun's rights in the underlying Java technology.²⁸⁴ The preservation of the intellectual property rights in improvements helps to stimulate innovations; however, the need to submit improvements to Sun for approval may delay or otherwise impede the implementation of new server/browser improvements, thereby detracting from the incentive to innovate.²⁸⁵

The current litigation between Sun and Microsoft over the lawfulness of the latter's modifications of Java illustrates the potential for conflict between one company's interest in maintaining a standard format and another company's interest in bringing its technological innovations to market. At base, however, a model that emphasizes the approval of technological changes either in advance or at the earliest possible date and a model in which the official standards evolve to incorporate previously implemented technological changes may, in the long run, produce similar results. In both models, these advances are, at some time, reconciled with industry standards.

One should expect that when Microsoft implements its latest technological advance in an effort to gain an edge on Netscape, the latter will complain that Microsoft is attempting to undermine universal access to the web. As servers and browsers incorporate new technologies, preexisting browsers will be unable to access material embodied in the new form.²⁸⁶ Accordingly, internet servers that make immediate use of new technology will be inaccessible in whole or in part to the older browsers.²⁸⁷ To that extent, universal access will indeed be impaired.²⁸⁸ This, however, is the way technology advances in a no-holds barred competitive setting like today's browser wars. In any event, the incompatibility will likely remain a temporary phenomenon that will disappear as protocols are modified and as new browser/server releases by both companies eventually overcome the incompatibilities generated by each step in the continuing technological advance.

284. See *Technology License and Distribution Agreement Between Sun Microsystems, Inc. and Microsoft Corporation* § 2.10(b) (visited Feb. 25, 1999) <<http://java.sun.com/lawsuit/document.html>> (agreeing that "Licensee retains all [intellectual property] rights, title and interest in and to" Java derivative or independent work).

285. The longer is the period of evaluation preceding use of the improvement, the more is the incentive to improve dampened.

286. See Lohr, *supra* note 271, at D1 (finding "[b]rownout troubles are partly an ironic byproduct of innovation").

287. See *id.* (explaining that as Microsoft and Netscape take divergent developmental pathways, they must tailor their software to make use of new innovations).

288. See *id.* (finding that older browsers are inaccessible to gamut of innovations).

There is, however, another route that innovation may take, a route that generates technology whose incompatibilities are not temporary. Either Microsoft or Netscape may opt for technology that is permanently incompatible with that of the other. Microsoft's J/Direct programming and its ActiveX software interface technology, for example, work only with Windows systems.²⁸⁹ Servers using these tools, therefore, would not be able to communicate with non-Windows systems. The roots of these incompatibilities lie deep in their structures and, as a result, are permanent. Innovations of this type are discussed below.

2. *Innovation Implicating the Java-Platform Paradigm*

Because standard Java must go through a process of conversion from Java byte code into machine code, Java runs slower than programs written to operate directly on the user's platform.²⁹⁰ One way of speeding up the operation of programs written in Java is to write to interface directly with the platform's APIs, avoiding the conversion process.²⁹¹ Java programmers have always had the option of invoking native code, but that option sacrifices Java's interoperability feature.²⁹² Sun's Java Development Kit 1.1 had native method invocations (NMIs) and its Java Development Kit 1.2 provides an improved native invocation called a Java Native Interface (JNI).²⁹³

Microsoft has attacked the problem by providing its own tools to enable Java programmers to invoke the APIs of the Windows system.²⁹⁴ Microsoft's J/Direct provides a Windows-specific interface.²⁹⁵ Microsoft's new Visual J++ supports J/Direct and a library of Java classes called Win-

289. See Jesse Berst, *Forecasting the Browser Battle*, WINDOWS SOURCES, Oct. 1997, at 47, 47 (discussing allegations that Microsoft uses J/Direct, which only functions with Windows because Microsoft wants to perpetuate software incompatibility); Martin Heller, *Java Explodes*, WINDOWS MAG., Aug. 1998, at 251, 251-52 (discussing J/Direct's Windows-only use); *TECHNIQUES: Programming for the Internet*, EXE, May 1, 1998, at 33, 33 (stating that ActiveX only functions with Windows and will only allow internet access if Microsoft's Explorer is used).

290. See Elliot, *supra* note 137, at 66 (reporting that JVM interpretation slows Java); Halfhill & Gallant, *supra* note 60, at 62-63 (noting that there are several factors that slow Java's performance, including dynamic binding and object hierarchies, hardware abstraction, interpreters that translate byte code instructions into native CPU instructions, automatically checking for array-bounds exceptions, multithreading, dynamic loading and restrictive security measures).

291. See Halfhill & Gallant, *supra* note 60, at 68 (stating that J/Direct offers "higher performance and richer functionality").

292. See *id.* (noting that "Java has always allowed" for invocation of native code).

293. See *id.* (stating that Java will introduce new version of its Java Development Kit (JDK) that will "introduce a better [native method invocation] called the Java native interface (JNI)").

294. See *id.* (noting Microsoft's offering of products for invoking APIs of Windows system).

295. See *id.* ("Microsoft's visual J++ supports a Windows-specific interface called J/Direct.").

dows Foundation Classes.²⁹⁶ The latter enables a programmer to wrap major parts of Windows APIs in Java code, thereby facilitating the connection between the program and the Windows system.²⁹⁷ Microsoft's development of these variations on Java has engendered a conflict with Sun, because Sun sees this technology as a threat to the integrity of Java.²⁹⁸ That threat does not appear to arise from the invocation of native code and the sacrifice of interoperability, however, because Java has long countenanced that option.²⁹⁹ Rather, the threat seen by Sun appears to lie in the implementation of technology that Sun has not approved. J/Direct and Windows Foundation Classes operate significantly faster than standard Java.³⁰⁰ They thus appear to meet a significant need. If, and to the extent that, they also operate faster than alternative methods for invoking native code, they may pose a threat to the Sun-approved version of Java.

Microsoft has also developed a set of software-to-software interfaces known as ActiveX.³⁰¹ ActiveX will perform many of the feats of Java, such as facilitating the production of small programs for activation on client machines (like the Java applets).³⁰² ActiveX, however, lacks standard Java's interoperability characteristic.³⁰³ Both the J/Direct technology and ActiveX outperform standard Java in speed.³⁰⁴ Microsoft's Internet Explorer 4.0 supports both J/Direct and ActiveX.³⁰⁵

296. See *id.* (describing Microsoft's Windows Foundation Classes (WFC), which lets one access library of Java classes).

297. See *id.* (stating that WFC "wrap[s] major parts of the Windows API in Java code that handles native method calls automatically").

298. See *id.* (noting that critics say Microsoft's strategies may threaten Java's integrity).

299. See *id.* (stating that Java has always allowed invocation of native code).

300. See *id.* (stating that "Microsoft's informal tests" show that "a WFC window with 23 miscellaneous components opens and closes about 600 times faster than an identical AWT window").

301. For a further discussion of the development of ActiveX, see *supra* notes 145-50 and accompanying text.

302. See Robert D. Hof, *Java's Cup Runneth Over*, BUS. WK., May 20, 1996, at 103 [hereinafter Hof, *Java's Cup*] (describing ActiveX creation of applets); Savage, *supra* note 147, at 80 ("ActiveX allows . . . programmers to easily include network functionality in existing programs.").

303. See Savage, *supra* note 147, at 82 (describing fact that Java, unlike ActiveX, operates on practically all types of operating systems).

304. See David S. Linthicum, *Java and ActiveX*, BYTE, Sept. 1997, at 61, 61 [hereinafter Linthicum, *Java*] (explaining that Windows's use of native features, such as J/Direct's machine code, causes ActiveX to operate much faster than Java).

305. See David S. Linthicum, *Detailing and Debating ActiveX*, COMPUTER SHOPPER, Oct. 1997, at 634, 634 [hereinafter Linthicum, *ActiveX*] (discussing fact that ActiveX only functions when using Internet Explorer). Microsoft announced at mid-Summer 1997 that, starting in July, J/Direct capability would augment its virtual machine for Java. See Sharon Gaudin, *Microsoft Tries to Reel in Java Jumpers Redmond to C++ Developers: Write Java, But for Windows*, COMPUTERWORLD, July 7, 1997, at 20, 20 [hereinafter Gaudin, *Write Java*] (announcing Microsoft's plans to incorporate J/Direct). Programmers would have the option of employing standard Java with full interoperability or J/Direct that would tie Java-built applications

Thus, for the operation of Windows-only networks, J/Direct and ActiveX may be superior to standard Java because they possess an advantage in speed and possibly in reliability.³⁰⁶ Although they lack Java's interoperability, interoperability is not relevant to a homogeneous network such as a Windows-only system where all users employ the same platform. Such Windows-only networks might be found among the intranets of particular business firms.³⁰⁷ A number of intranets have opted for one or both of these technologies.³⁰⁸

Java's promise of interoperability can be threatened when programmers for internet servers weigh Java's advantage in interoperability against its disadvantage in speed. If a programmer considers speed and/or reliability in interfacing with the Windows segment of the market more desirable than reaching all segments of the market, that programmer will likely prefer a method of interfacing directly with the Windows APIs. In such a case, the use of ActiveX technology or the J/Direct interface would be attractive options. If significant numbers of programmers begin to make the same judgments, the use of ActiveX or J/Direct would increase. As more programmers opt for ActiveX or J/Direct, Windows users would increasingly adopt the Explorer as their browser of choice. As the choice of ActiveX or J/Direct server programming grew, the incentives for non-Windows users to switch to a Windows platform would also increase.

This scenario bears the marks of network effects and path dependency theory. Yet, in this case especially, prudence cautions against drawing hasty conclusions. Proponents of network effects theory have stressed that network effects can result in the dominance of inferior technology, merely because the dominant technology jumped into an early lead as a result of some accident or other trivial event.³⁰⁹ As a result, rival—and superior—technologies were never able to catch up.³¹⁰ In other words, the technical superiorities of the rival technologies never sufficiently offset the network values appropriated by the technically inferior leader. Even

directly to the Windows platform. *See id.* at 20 (describing benefits of Microsoft's plan to incorporate J/Direct functions into its Java platform).

306. *See* Linthicum, *ActiveX*, *supra* note 305, at 634 ("[Java's] portability comes at the price of performance, since Java requires an interpreter between the code and the underlying operating system. ActiveX, by contrast, is really a Windows-only solution, but ActiveX controls offer native appearance and performance.").

307. *See* Preston Gralla, *Applets*, WINDOWS SOURCES, Mar. 1997, at 161, 161 (reporting that many intranet developers write applications for Windows-only network).

308. *See id.* (noting that some intranets have both forms of technologies).

309. *See* ARTHUR, *supra* note 168, at 24 (stating that "in practice, an early-start technology may already be locked in, so that a new arrival, although potentially superior, cannot gain a footing"); Arthur, *supra* note 152, at 116 (finding that "[w]hen two or more . . . increasing-return technologies 'compete' . . . for . . . a 'market' of potential adopters, insignificant events may by chance give one of them an initial advantage in adoptions").

310. *See* Arthur, *supra* note 152, at 116 (warning that allowing particular technology to "corner[] the market" may exclude other potentially better technology).

the most enthusiastic proponents of network effects, however, readily admit the possibility that the leader in a positive feedback market could possess technical superiority.³¹¹ Indeed, the leader in such a market does not inevitably attain its leadership as a result of a trivial event; the leader might have attained that leadership precisely because of the technical superiority of its product.³¹² The contribution of network effects theory merely shows that the leader could be, but is not necessarily, technically inferior.³¹³ Network effects theory helps us to focus on the gap—if any—between technical superiority and the value of network affiliation, and thus on the challenge that later entrants must overcome in challenging the dominant firm in a market exhibiting network effects.³¹⁴

In the above scenario, programmers begin to write in J/Direct or ActiveX because they prefer the speed and reliability associated with those tools over the universal accessibility of the present internet structure, including the present and emerging Java tools associated with that network. This is not a scenario of technical inferiority winning out over superior technology because of the value associated with belonging to a network. Rather, in this scenario, programmers opt for technical superiority (speed and reliability) over belonging to a network, *i.e.*, the present almost universally accessible internet network (including its Java tools).

As more programmers opt for ActiveX or J/Direct (rather than standard Java), the browser(s) that carry ActiveX and J/Direct capability increase in attractiveness to users because the server program base using these tools has increased.³¹⁵ Thus, there is a network effect—the server side of the network employing these tools has grown, resulting in the increased attractiveness of browsers carrying compatible technology.³¹⁶ As this increased attractiveness of technologically compatible browsers results in the use of more such browsers, the incentive for server programmers to employ ActiveX and J/Direct increases. There are network effects at work here, but the network effects are working in a significantly different way from the classic network effects scenario.³¹⁷

311. *See id.* (acknowledging possible adoption of more advanced technology).

312. *See* Katz & Shapiro, *Systems Competition*, *supra* note 152, at 108 (“[T]he market may be biased in favor of a new, superior, but incompatible technology . . .”).

313. *See id.* (noting real possibility that, despite incompatibility problems, more valuable technology will be most widely adopted).

314. *See* Gifford, *Microsoft Corporation*, *supra* note 33, at 641 (“[N]etwork effects reinforce the advantage secured by the market leader from its declining average cost of production. The producer who is ahead of its rivals . . . has advantages which are increasingly difficult for its rivals to overcome.”).

315. *See* Liebowitz & Margolis, *supra* note 152, at 133 (explaining how “utility” of product may increase as others also use such product).

316. *See id.* (finding that utility of product depends upon use of product).

317. For a further discussion of the classic network effects scenario, *see supra* notes 152-65 and accompanying text.

The Netscape Communicator, the Microsoft Internet Explorer 4.0 and the Sun Microsystems HotJava contain Java virtual machines, enabling them to receive and transmit to users standard Java programming.³¹⁸ As noted earlier, the Explorer's virtual machine uses a Java just-in-time compiler that speeds up the transformation of Java byte code into machine code. The Explorer's Java JIT compiler, together with the Java virtual machines of the Netscape and Sun browsers, critically affects the analysis of the network effects of ActiveX and J/Direct.³¹⁹ Server programmers need not choose between a superior technology with a small base and an inferior technology with a large base. If Java is the superior technology, server programmers will employ it, thereby serving the entire universe of Windows, many of which use the Explorer and its Java JIT compiler, plus Macintosh, OS2, Linux and UNIX platforms.³²⁰ If ActiveX or J/Direct are superior technologies, server programmers will weigh the advantages of the superior technology against the smaller installed base capable of handling it. In this case, if the programmers decide to employ ActiveX or J/Direct, they will be opting for superior technology over the value of adhering to the larger network.

Because programmers are not forced to choose between the universal Java network on one hand and the technology of ActiveX and J/Direct on the other, their choice to employ either of the latter two technologies shows that, in their view, the superior quality of these technologies outweighs the advantages of participating in the universal network. Network effects influence their choice as increasing numbers of servers employ ActiveX or J/Direct for technical reasons, and more users find the Explorer attractive as a result. Thus, the disadvantages of connecting to a small network are reduced, making the decision to use the superior technology easier. But at no point in this scenario are the programmers' decisions to opt for ActiveX or J/Direct pressured by network effects to substitute an inferior technology for a superior technology.

There is one caveat to this analysis. Although programmers presently use Java, it has not yet attained its full potential. Currently, Java provides small programs (applets) to internet users. The large software programs that can potentially be written in Java are only now beginning to be written. The Java platform paradigm, in which a Java-based browser replaces Windows as the user interface, has not yet arrived. A server programmer who today chooses J/Direct or ActiveX on the basis of speed and/or reliability over universal access is making that choice before Java has flowered

318. See Dragan & Seltzer, *supra* note 51, at 101 ("Web browsers such as Microsoft Internet Explorer, Netscape Navigator, and Sun's own HotJava have Java [virtual machines] built-in.").

319. For a further discussion of Java JIT compilers, see *supra* notes 141-43 and accompanying text.

320. See Gaudin, *Write Java*, *supra* note 305, at 20 ("The move to Java language is going to occur. [Microsoft] will support interoperability between Java and Windows They don't want developers to have to make a choice").

into its full potential. That choice reflects the programmer's weighing of the present advantages and disadvantages of J/Direct and/or ActiveX against the Java of today, not against the value of Java as it might be augmented by the technical improvements of tomorrow.

So long as all browsers maintain Java capability, the choice of programmers today to use Windows-only technology (such as J/Direct or ActiveX) does not foreclose a later move to Java when the problematic technical characteristics of Java (speed and reliability) are improved. Indeed, the Java capability of the three major browsers preserves Java's present potential and preserves the potentiality of Java to replace Windows-only technology at any time in the future when the relative technical advantages of the Java technology make such a move desirable.

Should Microsoft change the characteristics of the Explorer at some future time so that it could no longer handle standard Java, an entirely new situation would result. In that event, programmers would be forced to choose between the advantages of reaching the Explorer's installed base and the remainder of users (whose browsers would receive Java and other programming). Such a change in the Explorer's capabilities, however, would be economically irrational until the Explorer achieved dominance. Unless and until the Explorer achieved browser dominance, server programmers would write for the largest network—assuming, of course, that the technical advantages of Windows-only technology were insufficient to overcome the values of that network.³²¹ By hypothesis, the largest network would be the Java network. To remain competitive with the Netscape Navigator, the Explorer would have to access the majority of server sites.

Suppose, however, that the Explorer becomes the dominant browser. What could then stop Microsoft from changing the Explorer's capabilities to make it incompatible with standard Java? If Microsoft were to do this, it would face several problems. First, the installed base of all browsers, including Microsoft, Netscape and Sun browsers, is Java-compatible. All current browsers are equipped with Java interpreters and the Explorer itself has a JIT compiler. So long as most programmers were using Java, Microsoft could not start releasing new browsers incompatible with standard Java. If it did so, most users would reject the new edition of Explorer. Thus, to effectively change the Explorer's capabilities, Microsoft would have to do so all at once. It would have to "upgrade" the existing base of Explorers to a new version incompatible with Java. Could it do so? Probably not. Most Explorer users would probably decide not to upgrade. Indeed, the core of computer users who are highly conscious of technical

321. This assumption is necessary to the scenario. If Windows-only technology were sufficiently superior to offset the values of the Java network, then the Explorer and Windows-only technology would replace the Java-based network. On those facts, however, the movement from a Java-based network to a new Windows-only network would not be the result of network effects, but of the triumph of a superior technology.

developments and active in internet communication would alert others to the problematic nature of the "upgrade." In such circumstances, the mass media would likely publicize the matter, thereby informing even the most computer-illiterate users about the imminent threat to their access to internet material. In short, the scenario in which Microsoft flaunts its power over programmers is implausible.

More plausibility attaches to a scenario in which Microsoft—whether before or after it dominates the browser market—begins to make minor shifts in the technology of its browsers, maintaining backward compatibility with standard Java and other existent technology, but introducing improved capabilities on new browser editions. As a result, programmers could include enhanced programming that only the Explorer could read. They would not be required to include such programming because they would be able to reach all users, including those using the newest Explorer release, with standard Java and HTML. Should significant numbers of programmers decide to include additional features that only the Explorer could read, the attractiveness of the Explorer to users would increase. The resulting expansion of the Explorer's installed base would, in turn, further increase the attractiveness of the Explorer's unique technology to the programmers. But this scenario is one that we have already analyzed. The network effects in this scenario are subordinate to the technical factors. If the Explorer expands its share of the browser market as a result of the technical shifts introduced into the browser, product superiority, not network effects, would cause the expansion.

B. *Antitrust, Innovation and the Java Platform Paradigm*

It is possible that competition to sell browser/server software for use in intranets is focusing innovation upon speed and reliability.³²² The resulting software thus may not be interoperable because speed and reliability take precedence over interoperability in the intranet market.³²³ Should innovation, engendered by competition in the intranet market, produce browser/server software that so excels in speed and reliability that operators of internet servers opt for those qualities over interoperability, then interoperability will not be achieved through the Java platform. In that case, the possibility of a Java platform paradigm will have been eclipsed by a Windows platform paradigm.

322. See Matthew Phair, *Web Software Raises User Expectation, Presses Vendors*, ENGINEERING NEWS-REC., June 30, 1997, at 12, 12 ("Driven by users who enjoy the ease and speed of using browser software to access the latest information from anywhere in the world, vendors are being pressed to make their products Web-enabled for project networks, websites, databases and intranets.").

323. See Linthicum, *ActiveX*, *supra* note 305, at 634 ("[Java's] portability comes at the price of performance, since Java requires an interpreter between the code and the underlying operating system. ActiveX, by contrast, is really a Windows-only solution, but ActiveX controls offer native appearance and performance.").

A further variation on innovation cannot be ignored. Microsoft could concentrate its innovative efforts on improving the speed and reliability of Windows-only technology, such as ActiveX and J/Direct, to defeat the prospect of the Java platform paradigm replacing the Windows platform. That kind of activity could be described as innovation designed to preserve its Windows platform monopoly. But so long as the successes or failures of those technologies are determined in the open market, the motivations underlying innovative activity ought not to be a concern of the antitrust laws.

The purpose of the antitrust laws is not to impose a particular platform on the personal computer industry. Rather, it is to see that the market determines what that paradigm is to be. In the event that the Windows platform prevails, it will not be because network effects impose an inferior technology on computer users. Instead, it will be because users prefer the speed and reliability of the Windows-related software to the interoperability feature of Java.

VI. CONCLUSION

The complex and perplexing antitrust issues raised explicitly or implicitly in the Government/Microsoft and Sun/Microsoft litigation and in the disputes and rivalries between Microsoft and Netscape confuse even skilled antitrust lawyers. This Article has attempted to make a modest contribution towards unraveling the confusion and identifying the factors that underlie industry developments.

This Article is, of course, not the last word, not only because it addresses only some of the issues connected with the development of internet technology, but also because others will have additional analytical contributions. This Article has attempted to show, however, that network externalities, while an ever-present factor in the computer software industry, is not always the critical factor determining the course of development. Indeed, if innovations in server/browser technology from Microsoft win widespread support in the internet marketplace, this may be an example of a superior technology triumphing over network effects.

